

Universidad de las Ciencias Informáticas

Facultad 1

Título: Ingeniería de Requisitos para Xerberos.

**Trabajo de Diploma para optar por el título de
Ingeniero Informático**

Autora: Mairim Delgado Muñiz.

Tutoras: Ing. Gladys Marsi Peñalver Romero

Ing. Yusleydi Fernández del Monte

La Habana, Cuba, Junio 2012.

“Año 54 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora de este trabajo y autorizo al centro CESOL de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los 18 días del mes de junio del año 2012.

Mairim Delgado Muñiz.

Ing. Gladys Marsi Peñalver Romero.

Ing. Yusleydi Fernández del Monte.

RESUMEN

Como parte del proceso de informatización de la sociedad cubana surge Xerberos, alternativa libre que soluciona el problema de la gestión de estaciones en cualquier empresa u organización que presente un número considerable de estaciones de trabajo. El producto surge como sucesor de Sistema de Clonación de Imágenes y Sistemas Operativos (SistClon) con el objetivo de implementar un software con más funcionalidades y una mejor arquitectura. Dentro de las prioridades fundamentales del nuevo sistema por sus características distintivas se encuentra realizar una correcta captura y gestión de requisitos para garantizarle al producto final una calidad indiscutible, ya que evita uno de los grandes problemas, la mala gestión de los mismos. Con el objetivo de brindar una solución se propone un conjunto de procesos de Ingeniería de Requisitos a seguir a lo largo del ciclo de vida de desarrollo del producto que permita la calidad óptima desde sus inicios. El conjunto de procesos obtenidos fue evaluado mediante el método de predicción Delphi, el cual garantizó la completitud y adecuación de cada uno de ellos para propiciar un correcto desarrollo del software Xerberos.

PALABRAS CLAVE: Ingeniería de Requisitos, SistClon, Xerberos.

Indice

<u>INTRODUCCIÓN.....</u>	<u>1</u>
<u>CAPÍTULO 1: CONCEPTOS BÁSICOS PARA LA INVESTIGACIÓN.....</u>	<u>6</u>
<u>Principales Conceptos.....</u>	<u>6</u>
<u>De la IR.....</u>	<u>7</u>
Características de los requisitos.....	8
Priorización de los requisitos.....	9
Documento Especificación de Requisitos.....	9
Importancia de la IR.....	9
Procesos de IR.....	10
<u>Modelos, Estándares y Normas.....</u>	<u>14</u>
Modelo CMMI.....	14
Proceso de Mejora basado en el Modelo CMMI.....	16
Normas.....	17
Estándares.....	18
<u>Técnicas y herramientas.....</u>	<u>18</u>
Técnicas a utilizar en los procesos de IR.....	18
Herramientas.....	21
<u>Xerberos.....</u>	<u>25</u>
<u>CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.....</u>	<u>28</u>
<u>PDSW de Xerberos.....</u>	<u>28</u>
<u>Descripción de los PIR propuestos.....</u>	<u>31</u>

Inicio del proyecto.....	31
Obtención de requisitos para la construcción del sistema.	33
Análisis de requisitos.	35
Análisis para la concepción de las historias de usuario.....	35
Análisis para el desarrollo de los módulos del sistema.	36
Análisis del desarrollo de la interfaz para administrar el sistema.....	39
Análisis de requisitos.....	40
Especificación de requisitos.....	41
Verificación de requisitos.....	43
Gestión de requisitos.....	46
Cronograma a seguir para implementar los PIR.....	48
<i>CAPÍTULO 3: EVALUACIÓN DE LOS PIR EN EL SISTEMA XERBEROS.....</i>	<i>50</i>
El panel de expertos.....	50
Delphi como método de experto.....	50
Características del método Delphi:.....	50
Elección de los especialistas.....	51
Determinación del coeficiente de competencia de los especialistas.....	52
Elaboración del cuestionario.....	55
<i>CONCLUSIONES.....</i>	<i>65</i>
<i>RECOMENDACIONES.....</i>	<i>66</i>
<i>REFERENCIAS BIBLIOGRÁFICAS.....</i>	<i>67</i>
<i>BIBLIOGRAFÍA.....</i>	<i>71</i>
<i>ANEXO.....</i>	<i>76</i>

<u>Anexo1.....</u>	<u>76</u>
<u>Anexo2.....</u>	<u>77</u>
<u>Anexo3.....</u>	<u>78</u>
<u>GLOSARIO DE TÉRMINOS.....</u>	<u>88</u>

INTRODUCCIÓN

En los últimos años, el proceso de desarrollo de software (PDSW) se ha convertido en uno de los principales indicadores en la industria de la informática. En este proceso las malas prácticas de la Ingeniería de Software traen como consecuencia el incumplimiento de los compromisos productivos, ocasionando altos costes tanto para el equipo de desarrollo como para el cliente, por ello, se impone como disciplina a seguir durante todo el ciclo de vida de un proyecto de software.

Dentro de esta disciplina se encuentra como área más activa: la Ingeniería de Requisitos (IR) que brinda principios, métodos, herramientas y técnicas para descubrir y documentar los requisitos [1]; ínfima parte del PDSW que se hace imprescindible para obtener un producto final con la calidad requerida por los clientes o usuarios finales. Los requisitos marcan el punto de partida para actividades como la planeación, la estimación de tiempos y costos, la definición de recursos necesarios y la elaboración de cronogramas, principales mecanismos de control con los que se cuenta durante la etapa de desarrollo.

En el mundo, los procesos de esta área se definen desde el inicio en cualquier tipo de sistema porque se enfocan en un área fundamental: la definición de lo que se desea producir. Mediante los mismos se puede verificar si se alcanzaron o no los objetivos establecidos en el proyecto, ya que brindan un modelo a seguir para no cometer errores.

La calidad no puede implementarse en las etapas finales de los procesos de desarrollo sino que es una característica intrínseca al propio producto, iniciándose en las especificaciones de los productos. Este inicio de las actividades de calidad es la base para asegurar el perfeccionamiento del proyecto.

En la Universidad de las Ciencias Informáticas (UCI), el desarrollo de software se encuentra en constante ascenso. Este proceso actualmente se organiza en varios centros, los cuales se encargan de desarrollar productos enfocados a diferentes perfiles de la informática.

En particular, el Centro de Software Libre (CESOL) perteneciente a la Facultad 1 se identifica con el perfil de migración hacia aplicaciones de código abierto, soporte técnico y publicidad. En la actualidad uno de los mayores problemas, enfrentados por cualquier institución o empresa que presente al

menos un número considerable de estaciones de trabajo es el servicio de un sistema que brinde una administración centralizada y flexible. Por esta razón y como parte del proceso de informatización de la sociedad cubana, se propone desarrollar la suite Xerberos como alternativa libre para la automatización del proceso de soporte, seguridad y mantenimiento a estaciones de trabajo de forma rápida, eficaz y de fácil dominio para los administradores o el personal encargado de esta tarea.

Xerberos es un sistema que permite detectar amenazas de seguridad como el uso indebido de los recursos de la red, acceso ilegal a estaciones o terminales, daños maliciosos y fuga de información sensible. Integra funcionalidades para auditar acciones, procesos y recursos dentro de la empresa, brindando notificaciones de posibles modificaciones y actividad indebida en los mismos. También permite realizar actividades de soporte y mantenimiento en estaciones de trabajo gracias a un sistema de clonación y administración remota de imágenes de sistemas operativos, permitiendo gestionar de forma centralizada las particiones, las configuraciones, instalación y desinstalación de software en cada estación. Su interfaz amigable y sencilla permite un fácil acceso a cada funcionalidad del sistema y a los reportes [2]. Para su desarrollo se han definido diferentes módulos, entre los cuales, actualmente se han obtenido versiones inestables del módulo de Clonación y de Comunicación.

El producto Xerberos surge como sucesor de SistClon con el objetivo de implementar un software con más funcionalidades y una mejor arquitectura. Xerberos presenta características distintivas tales como:

- No posee un cliente para la captura de requisitos.
- Los requisitos son incorporados a partir de investigaciones a sistemas similares.
- Los requisitos son refinados a partir de estudios realizados por parte del equipo de desarrollo.
- Posee un pequeño equipo de trabajo.

Dentro de los problemas identificados en su PDSW pueden mencionarse la no realización de una correcta captura de las necesidades para convertirlas luego en los diversos requisitos funcionales, no funcionales y sus respectivas especificaciones; la no existencia de artefactos o procesos que permitan: la trazabilidad de los requisitos tanto hacia delante como hacia atrás y la gestión de los

mismos. Lo anteriormente mencionado trae como consecuencias, graves problemas en la obtención del producto final con calidad, pues el mal trabajo con los requisitos desde el inicio ocasiona: altos costes para el proyecto, incumplimientos en las metas trazadas e insatisfacción por parte de los usuarios finales. Partiendo de la presente problemática se define como **problema científico**: ¿Cómo aplicar la IR en el proceso de desarrollo de Xerberos?

Para dar solución al mismo se define como **objetivo general de la investigación**: elaborar un conjunto de procesos que garanticen la gestión de requisitos en el desarrollo del sistema Xerberos.

Para darle cumplimiento al objetivo general se definieron los siguientes **objetivos específicos**:

- Investigar los procesos que componen la IR.
- Investigar como se lleva a cabo la IR en el desarrollo de sistemas similares a Xerberos.
- Elaborar los procesos de IR.
- Evaluar los procesos propuestos.

Los anteriores objetivos se concretan en las siguientes **tareas investigativas**:

- Investigación de las buenas prácticas de los procesos de IR que se desempeñan nacional e internacionalmente, para conocer con profundidad el área de conocimientos en la que se trabaja.
- Investigación acerca del proceso de desarrollo en el sistema Xerberos, para lograr una correcta gestión de requisitos en el mismo.
- Identificación de los procesos de IR a aplicar en el sistema Xerberos en correspondencia con sus características, para lograr un correcto desempeño de la IR.
- Diseño de procesos de IR para el sistema Xerberos para contribuir a elevar la calidad de este producto.
- Evaluación de los procesos propuestos en el sistema Xerberos, verificando así los beneficios que aportan los procesos diseñados para el desarrollo del sistema.

Definiéndose como **objeto de estudio** de la investigación: Los procesos de IR, enmarcando como **campo de acción**: Los procesos de IR en el sistema Xerberos.

La investigación se sustenta en la siguiente **idea a defender**: Las buenas prácticas de la IR en el proyecto Xerberos puede facilitar el desarrollo de software y el perfeccionamiento de la calidad.

Para un mejor desarrollo de la investigación se utilizaron los siguientes **métodos científicos**:

Inductivo-deductivo: La inducción se puede definir como una forma de razonamiento por medio de la cual se pasa del conocimiento de casos particulares a un conocimiento más general que refleja lo que hay de común en los fenómenos individuales. La deducción es una forma de razonamiento, mediante la misma se pasa de un conocimiento general a otro de menor nivel de generalidad. Se aplica en el estudio de los procesos utilizados en sistemas similares a Xerberos para obtener una visión de este tipo de software.

Analítico–Sintético: Permite la división mental del todo en sus múltiples relaciones y componentes, establece mentalmente la unión entre las partes, previamente analizadas y posibilita descubrir relaciones y características generales entre los elementos de la realidad. Se utiliza este método en la investigación de los procesos de IR existentes y en la comprensión de cada uno de ellos para definir cuáles darán solución al problema.

Análisis Histórico-Lógico: Plantea que los fenómenos no suceden al azar, sino que son producto de un proceso que los origina, motiva o simplemente da lugar a su existencia. Este método es usado en el estudio del surgimiento de la IR así como su evolución.

La investigación está compuesta por: la introducción, tres capítulos, las conclusiones, las recomendaciones, la bibliografía y los anexos; la estructura de los capítulos se describe a continuación:

- **Capítulo 1**: Conceptos básicos para la investigación. En el capítulo se expondrán los principales conceptos y aspectos relacionados con la IR. Además, se abordan las fases del PDSW de Xerberos según la metodología de desarrollo.
- **Capítulo 2**: Procesos de Ingeniería de Requisitos a aplicar en Xerberos. En el capítulo se describen los procesos de IR que se van a aplicar en Xerberos. Además, se define, por las

fases del proceso de desarrollo del sistema, el orden en que se deben ejecutar los procesos y cómo hacerlo.

- **Capítulo 3:** Evaluación de los procesos de Ingeniería de Requisitos en el sistema Xerberos. En el capítulo se evalúan los procesos propuestos mediante el método de expertos Delphi.

CAPÍTULO 1: CONCEPTOS BÁSICOS PARA LA INVESTIGACIÓN

En el presente capítulo se precisan elementos teóricos que sustentan la investigación y el desarrollo del tema propuesto. Se analizan los principales elementos en lo que se refiere a la IR, así como los modelos, estándares y normas. También se describen las técnicas de captura de requisitos, herramientas y procesos de la misma según diferentes autores. Además, se realiza un acercamiento a la descripción de Xerberos y a la metodología que va a guiar su desarrollo.

Principales Conceptos

A continuación se presentan los principales conceptos relacionados con la investigación.

Requisitos: Los requisitos según la definición que aparece en el glosario de la Institute of Electrical and Electronics Engineers (IEEE) son:

- Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo.
- Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal.
- Una representación documentada de una condición o capacidad [\[3\]](#).

Pueden dividirse en requisitos funcionales y no funcionales. Los primeros definen las funciones que el sistema va a ser capaz de realizar y describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Los segundos tienen que ver con las características que de una u otra forma puedan limitar el sistema, entre ellos se encuentran, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares.

Ingeniería de Requisitos

“La IR ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, qué es lo que el cliente quiere y cómo interactúan los usuarios finales con el software” [\[4\]](#).

“La IR es el proceso de desarrollar una especificación de software. Las especificaciones pretenden comunicar las necesidades del sistema y del cliente a los desarrolladores del sistema” [5].

Luego de revisar las anteriores definiciones se puede definir que la IR es la disciplina que logra la definición y gestión correcta de los requisitos necesarios, precisos y concisos para lograr un producto final que cumpla con las expectativas y necesidades del cliente.

Proceso de Ingeniería de Requisitos

El proceso de IR se utiliza para definir todas las actividades involucradas en el descubrimiento, documentación y mantenimiento de los requisitos para un producto de software determinado, donde es muy importante tomar en cuenta que su aporte vendrá a ayudar a determinar la viabilidad de llevar a cabo el software (si es factible llevarlo o no) [6].

Trazabilidad de Requisitos.

Según la IEEE 830-1998, es la habilidad para seguir la vida de un requisito en ambos sentidos, hacia sus orígenes o hacia su implementación, a través de las especificaciones generadas durante el proceso de desarrollo. La trazabilidad de requisitos es considerada un proceso imprescindible para la adecuada gestión de requisitos [7].

De la Ingeniería de Requisitos

Es muy importante mencionar que el poder formular una especificación de requisitos completa y consistente, es un paso muy importante para evitar cometer errores en la definición de los mismos, pues puede resultar caro corregirlos una vez desarrollado el sistema. De ahí, la vital importancia que tiene en la IR generar una adecuada especificación que contemple claramente y sin ambigüedades las funcionalidades del sistema a desarrollar, con el fin primordial de evitar que los proyectos fracasen debido a una mala elaboración de la definición y especificación de requisitos [8]. Por la importancia que posee la correcta especificación de requisitos, a continuación se exponen las dificultades más recurrentes en la definición de los requisitos:

- Los requisitos no son obvios y vienen de muchas fuentes.
- Son difíciles de expresar en palabras (el lenguaje es ambiguo).

Ingeniería de Requisitos para Xerberos
CAPÍTULO 1: CONCEPTOS BÁSICOS PARA LA INVESTIGACIÓN

- Existen muchos tipos de requisitos y diferentes niveles de detalle.
- La cantidad de requisitos en un proyecto puede ser difícil de manejar.
- Nunca son iguales. Algunos son más difíciles, más riesgosos, más importantes o más estables que otros.
- Los requisitos están relacionados unos con otros, y a su vez se relacionan con otras partes del proceso.
- Cada requisito tiene propiedades únicas y abarcan áreas funcionales específicas.
- Un requisito puede cambiar a lo largo del ciclo de desarrollo.
- Son difíciles de cuantificar, pues cada conjunto de requisitos es particular para cada proyecto.

Características de los requisitos

Un conjunto de requisitos en estado de madurez debe presentar una serie de características, tanto individualmente como en grupo. A continuación se presentan las más importantes.

- **Necesario:** Un requisito es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
- **Conciso:** Un requisito es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- **Completo:** Un requisito está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- **Consistente:** Un requisito es consistente si no es contradictorio con otro requisito.
- **No ambiguo:** Un requisito no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.

- **Verificable:** Un requisito es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas [\[9\]](#).

Priorización de los requisitos

El enfoque común para la priorización de los requisitos puede verse en tres categorías. A continuación se muestran tres niveles de escalas, estas escalas son subjetivas e imprecisas, por lo que todos los involucrados deben ponerse de acuerdo sobre el significado.

- **Alto:** Un requisito de misión crítica, necesaria para la próxima versión.
- **Medio:** Apoya las operaciones necesarias del sistema, se requiere tiempo, pero podría esperar hasta una versión posterior si es necesario.
- **Bajo:** Una mejora funcional o de calidad, sería bueno tener algún día, si los recursos lo permiten [\[10\]](#).

Documento Especificación de Requisitos

Este documento recoge la especificación de requisitos así como toda la documentación correspondiente del análisis de la aplicación. A partir de este se establece un diseño que se ajuste a los requisitos expuestos, después de realizar el estudio de reutilización para determinar los componentes disponibles, que se ajustan a los mismos y que se puedan incorporar al diseño. Esta especificación se debe estructurar siguiendo la norma IEEE 830 en todos los apartados que sean posibles. El objetivo es la creación de un documento de especificación que sea completo y no contenga ambigüedades [\[11\]](#).

Importancia de la IR

Los principales beneficios que se obtienen de la IR son:

- Permite gestionar las necesidades del proyecto en forma estructurada: cada actividad de la IR consiste de una serie de pasos organizados y bien definidos.

- Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados: la IR proporciona un punto de partida para controles subsecuentes y actividades de mantenimiento, tales como estimación de costos, tiempo y recursos necesarios.
- Disminuye los costos y retrasos del proyecto: muchos estudios han demostrado que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro.
- Mejora la calidad del software: la calidad en el software tiene que ver con cumplir un conjunto de requisitos (funcionalidad, facilidad de uso, confiabilidad, desempeño).
- Mejora la comunicación entre equipos: la especificación de requisitos representa una forma de consenso entre clientes y desarrolladores. Si este consenso no ocurre, el proyecto no será exitoso.
- Evita rechazos de usuarios finales: la IR obliga al cliente a considerar sus requisitos cuidadosamente y revisarlos dentro del marco del problema, por lo que se le involucra durante todo el desarrollo del proyecto [\[12\]](#).

Procesos de IR

Los procesos de IR se describen y se ejecutan en el desarrollo de un sistema para establecer la comunicación de los involucrados en el proyecto, entender sus necesidades, analizarlas, especificar la solución sin ambigüedades, validar la especificación, y administrar los requisitos conformes estos se transforman en un sistema operacional [\[13\]](#). La IR es un proceso que comprende todas las actividades para crear y mantener los requisitos de un sistema. Según Roger S. Pressman comprende siete distintas funciones:

- **Inicio del proyecto:** El objetivo es establecer una comprensión básica del problema, las personas que quieren una solución, la naturaleza de la solución que se desea, y la efectividad de la comunicación preliminar entre el cliente y el desarrollador.
- **Obtención:** Se recopilan los requisitos tanto funcionales como no funcionales del sistema, determinando para ello el objetivo del mismo, cuáles serán sus características y de qué forma será utilizado. Usualmente los proyectos en esta fase enfrentan problemas que pueden resumirse, de ámbito: límites del sistema mal definidos; de comprensión: clientes/usuarios no seguros de que necesitan, no comprenden todo el dominio del problema, dificultades al

comunicar necesidades, omiten información que consideran obvia, especifican requisitos ambiguos o inestables; de volatilidad: los problemas cambian conforme transcurre el tiempo.

- **Elaboración:** La información conseguida con el cliente durante el inicio del proyecto y la obtención se expanden y se refinan durante la elaboración. Se enfoca al refinado de las funciones, características y restricciones del software. El resultado final es un modelo de análisis que define el dominio de la información, las funciones y el comportamiento del problema.
- **Negociación:** Se concilian los conflictos existentes entre requisitos por medio de un proceso de negociación. Mediante un enfoque iterativo, los requisitos se eliminan, combinan o modifican de forma que se obtenga por parte del cliente y los desarrolladores un grado de satisfacción.
- **Especificación:** La especificación es el producto de trabajo final que genera la IR. Sirve como base para las actividades de Ingeniería de Software subsecuentes. Describe la función y el desempeño de un sistema basado en computadoras y las restricciones que regirán su desarrollo.
- **Validación:** La validación examina la especificación para asegurar que todos los requisitos de software se han establecido de manera precisa, se han detectado inconsistencias, omisiones y errores y que estos han sido corregidos, y que los productos de trabajo cumplen con los estándares establecidos para el proceso, proyecto y producto. El mecanismo primario para la validación es la revisión técnica formal.
- **Gestión de Requisitos:** Es un conjunto de actividades que ayudan al equipo de proyecto a identificar, controlar y rastrear los requisitos y los cambios a estos en cualquier momento mientras se desarrolla el proyecto. La gestión comienza con la identificación, cada requisito se asigna a un solo identificador. Una vez identificados los requisitos se desarrollan las tablas de trazabilidad, cada una de ellas relaciona los requisitos con uno o más aspectos del sistema o de su ambiente [14].

Ian Sommerville presenta el proceso de IR mediante un modelo en espiral, como una actividad de tres etapas donde las actividades se organizan como un proceso iterativo alrededor de un espiral. La

cantidad de dinero y esfuerzo dedicado a cada actividad en una iteración depende de la etapa del proceso general y del tipo de sistema desarrollado. Al principio, se dedicará la mayor parte del esfuerzo a la comprensión del negocio, de los requisitos no funcionales y del usuario. Al final del proceso, en el anillo exterior de la espiral, se dedicará un mayor esfuerzo a la IR del sistema y al modelado de este. Este modelo satisface enfoques de desarrollo en los cuales los requisitos se especifican a diferentes niveles de detalles. El número de iteraciones en la espiral puede variar y se puede salir de la misma después que se hayan obtenido algunos o todos los requisitos del usuario [15].

Según Ralph R. Young las actividades de IR para el ciclo de vida de un sistema son:

- **La identificación de las partes interesadas:** Cualquier persona que tenga un interés en el sistema o en las cualidades que posee, que satisfagan sus necesidades particulares.
- **Aumentar la comprensión de los clientes y usuarios sobre las necesidades y expectativas del sistema:** Elicitación de requisitos, los mismos pueden incluir varios tipos.
- **Identificación de requisitos:** Indica los requisitos en sencillas sentencias y los ofrece en conjunto.
- **La clarificación y la reformulación de los requisitos:** Describe las necesidades reales del cliente.
- **El análisis de los requisitos:** Asegura que están bien definidos y que se ajusten a los criterios de un buen requisito.
- **Definición de los requisitos de una manera que signifique lo mismo para todas las partes:** Cada grupo de interés puede tener una perspectiva diferente del sistema y los requisitos del sistema.
- **Especificación de los requisitos:** Requiere incluir todos los detalles de cada requisito para que pueda ser incluido en un documento de especificación.
- **Dar prioridad a los requisitos:** Priorizar los requisitos proporciona la oportunidad de dirigirse a la más alta prioridad primero y posiblemente liberar una versión de un producto.

- **Derivando requisitos:** Algunos requisitos que surgen debido al diseño de un sistema, pero no proporcionan un beneficio directo para el usuario final.
- **Clasificando requisitos:** Se tipifican los requisitos los de: hardware, software, capacitación y documentación.
- **La asignación de los requisitos:** Se asignan los requisitos a los diferentes subsistemas, y componentes del sistema.
- **Seguimiento de los requisitos:** Se necesita la capacidad de rastrear o seguir dónde se cumple cada requisito en el sistema.
- **Gestión de requisitos:** Añadir, eliminar y modificar requisitos durante: todas las fases del diseño del sistema, el desarrollo, la integración, las pruebas, el despliegue y la operación.
- **Comprobación y verificación de los requisitos:** Verifica en cuanto a: requisitos, diseños, códigos, planes de prueba, y los productos del sistema, para asegurar que se cumplen los requisitos.
- **Validación de requisitos:** Confirma que los verdaderos requisitos se aplican en el sistema de entrega [\[16\]](#).

Modelos y Estándares

La ingeniería de software dispone de varios modelos, estándares y normas que contemplan la IR, en los cuales se apoya para la construcción del software. A continuación se abordan los más importantes.

Modelo CMMI

Capability Maturity Model Integration (CMMI) consiste en una serie de prácticas y procesos de alto nivel que ayudan a las organizaciones a construir un modelo para mejorar sus procesos. Está conformado por dos enfoques para la mejora de procesos, según la capacidad del proceso (representación continua) y según la madurez de la organización (representación escalonada).

Por la representación escalonada CMMI posee un conjunto de áreas de procesos, las que se dedican al tratamiento de requisitos son:

- La administración de Requisitos (*Requirements Management* (REQM)) enmarcada en el área de nivel 2 de madurez (Gestionado) donde su propósito es administrar los requisitos del proyecto e identificar inconsistencias entre dichos requisitos, los subproductos y planes del proyecto, además que los procesos estén planeados, ejecutados, medidos y controlados. Ésta área de procesos está conformada por cinco prácticas específicas (SP por sus siglas en inglés):

SP 1.1 Obtener el entendimiento de los requisitos: desarrollar con los proveedores de requisitos el entendimiento del significado de los requisitos.

SP 1.2 Obtener el compromiso a los requisitos: lograr el compromiso a los requisitos por parte de los participantes del proyecto.

SP 1.3 Administrar los cambios a los requisitos: administrar los cambios a los requisitos conforme vayan evolucionando en el proyecto.

SP 1.4 Mantener la trazabilidad bidireccional de los requisitos: mantener el traceo bidireccional entre los requisitos y los productos de trabajo.

SP 1.5 Identificar inconsistencias entre el trabajo del proyecto y los requisitos: identificar las inconsistencias entre los planes del proyecto, los productos de trabajo y los requisitos.

No poner en práctica REQM ocasiona: entendimientos diferentes entre clientes y desarrolladores sobre el conjunto de requisitos, esfuerzos del equipo de desarrollo en proyectos no factibles, inexistencia de un registro de cambios y sus respectivas justificaciones y la inseguridad acerca de la inclusión de requisitos originales con sus cambios aprobados.

- El desarrollo de los requisitos englobado en el área 3 del nivel madurez (Definido). Los procesos aquí se encuentran bien caracterizados y son estandarizados para toda la organización, con herramientas y métodos. Proporcionan consistencia a la organización. Su propósito en cuanto a requisitos es identificar, desarrollar, generar, determinar y analizar las

necesidades de los usuarios, las cuales se transforman en las características que tendrá el producto final.

En esta área de proceso, CMMI propone los siguientes objetivos y prácticas relacionadas:

- Desarrollar los requisitos del cliente. El término cliente acuña todos los perfiles involucrados en el desarrollo del software: desde los propios clientes, hasta usuarios finales pasando por proveedores. El objetivo es recoger toda la información necesaria para conseguir un conjunto de requisitos del cliente, donde las expectativas, restricciones, interfaces, conceptos operacionales y legales, estén presentes.
- Desarrollar los requisitos del producto. En el proceso iterativo se pasa del conjunto de los requisitos del cliente a un conjunto de requisitos del producto y de sus componentes. Este mapeo incluye procesos, objetos y personal; constituyendo la base para la solución técnica.
- Analizar y validar los requisitos. Una vez que se dispone de un conjunto de requisitos, estos deben ser analizados y validados para tener la garantía de que los requisitos son coherentes con las intenciones de los Casos de Uso y no hay conflictos entre ellos [\[17\]](#).

Proceso de Mejora basado en el Modelo CMMI

La UCI es un centro productivo, cuya misión es producir software y servicios informáticos a partir de la vinculación estudio–trabajo como nuevo modelo de formación. Es considerada la mayor organización productora de software en el país. En la actualidad dicho centro está acometiendo un proyecto de mejora de sus procesos basado en la obtención de la certificación internacional en el nivel 2 del modelo CMMI y con la contratación de los servicios de consultoría del *Software Industry Excellence Center (SIE Center)* del Tecnológico de Monterrey. Hecho que la convertirá en la primera empresa cubana certificada con este modelo.

CMMI es un modelo de referencia para el crecimiento de capacidades y madurez, que se enfoca tanto en procesos de administración como de ingeniería de sistemas y software. Con su instauración se espera alcanzar beneficios como: calendarios y presupuestos predecibles en los proyectos,

mejora del ciclo de vida dentro del desarrollo de software, mayor productividad, mayor calidad de los productos y servicios que ofrece la universidad a sus clientes y la satisfacción de los mismos.

El servicio de consultoría que ofrece el *SIE Center* permite:

- Ayudar a la UCI a revisar su estrategia de mejora de procesos de software, para asegurar que su organización está basada en procesos y con un programa de mejora continua, alineado con sus objetivos de negocio.
- Ayudar a la UCI a establecer las bases y fundamentos para seguir mejorando sus procesos y fortalecer su cultura de calidad en el desarrollo de software.
- Alinear los procesos de desarrollo de software con los principios y requisitos del modelo CMMI, estableciendo planes de mejora con los que la organización oriente sus procesos hacia la consecución de sus metas [\[18\]](#).

El proceso de mejora basado en el modelo CMMI está compuesto por varias áreas de procesos, de ellas la dedicada al análisis de los requisitos es la Administración de Requisitos. La misma define como artefactos las planillas de: Especificación de requisitos de software, Requisitos rechazados, Criterios para validar requisitos del cliente, Criterios para validar requisitos del producto, Evaluación de requisitos, Descripción de requisitos, Criterios para definir proveedores válidos de requisitos, Lista de chequeo para detectar inconsistencias en REQM y Reporte de trazabilidad.

Estándares

Los estándares son acuerdos documentados que contienen especificaciones técnicas u otros criterios precisos para ser usados consistentemente como reglas, guías, o definiciones de características para asegurar que los materiales, productos, procesos y servicios se ajusten a su propósito [\[19\]](#). Se identifican como útiles en la investigación actual:

- IEEE Std 830-1998 Prácticas recomendadas para especificaciones de requisitos de software, establece las guías para una buena especificación de requisitos, abordando su naturaleza, entorno, características, evolución, prototipado, integración en el diseño y los requisitos del proyecto. Propone además una estructura para el documento teniendo en cuenta los factores

antes mencionados y las políticas de calidad para el control de cambios y registros históricos del documento [20].

- IEEE Std 610.12-1990 Glosario Estándar para la Terminología de Ingeniería de Software, ofrece una ayuda para el común nombramiento en el lenguaje técnico (IEEE610, 1991) [21].
- IEEE Std 1012-1998 Estándar para la verificación y validación de software, ofrece sus consideraciones acerca de la garantía de calidad del producto en revisiones hacia lo interno en el cumplimiento de los requisitos del sistema y hacia lo externo en la satisfacción de las necesidades del cliente [22].

Técnicas y herramientas

La evolución de los estudios encarados por la IR se fue dando paulatinamente. Sin embargo, los esfuerzos se concentraron en la búsqueda de técnicas, métodos y herramientas que pudieran ser aplicados durante el proceso de definición de requisitos. A continuación se muestran las principales técnicas y herramientas utilizadas en la IR para la captura y gestión de los requisitos.

Técnicas a utilizar en los procesos de IR.

Cuando sea necesario, el analista va a emplear una combinación de estas técnicas para establecer los requisitos exactos con las personas implicadas, para producir un sistema que resuelva las necesidades del negocio. A continuación se muestran ejemplos de las técnicas más usadas y seguidamente las ventajas y desventajas de las mismas.

Entrevistas y Cuestionarios: Las entrevistas y cuestionarios se emplean para reunir información proveniente de personas o de grupos. Durante la entrevista, el analista conversa con el encuestado; el cuestionario consiste en una serie de preguntas relacionadas con varios aspectos de un sistema.

Tormenta de Ideas (Brainstorm): Este método comenzó en el ámbito de las empresas, aplicándose a temas tan variados como: la productividad, la necesidad de encontrar nuevas ideas y soluciones para los productos del mercado y encontrar nuevos métodos que desarrollen el pensamiento creativo a todos los niveles. Pero pronto se extendió a otros ámbitos, incluyendo el mundo de desarrollo de sistemas; básicamente se busca que los involucrados en un proyecto desarrollen su creatividad.

Prototipos: Comienzan con la captura de requisitos. Desarrolladores y clientes se reúnen y definen los objetivos globales del software, identifican todos los requisitos que son conocidos, y señalan áreas en las que será necesaria la profundización en las definiciones. Luego tiene lugar un diseño rápido, que consiste en la representación de aquellos aspectos del software que serán visibles al usuario (entradas, formatos de salidas). El diseño rápido lleva a la construcción de un prototipo. El prototipo es evaluado por el cliente y el usuario. Además, es utilizado para refinar los requisitos del software.

Proceso de Análisis Jerárquico (The Analytic Hierarchy Process (AHP)): Esta técnica tiene por objetivo resolver problemas cuantitativos, para facilitar el pensamiento analítico y las métricas. Consiste en una serie de pasos a saber:

- Encontrar los requisitos que van a ser priorizados.
- Combinar los requisitos en las filas y columnas de la matriz $n \times n$ de AHP.
- Hacer algunas comparaciones de los requisitos en la matriz.
- Sumar las columnas.
- Normalizar la suma de las filas.
- Calcular los promedios [\[23\]](#).

Historia de usuarios: Es una representación de un requisito de software escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollos ágiles para la especificación de requisitos (acompañadas de las discusiones con los usuarios y las pruebas de validación). Cada una debe ser limitada, ésta debería poderse escribir sobre una nota adhesiva pequeña [\[24\]](#).

Revisiones técnicas formales: Las revisiones técnicas se deben principalmente a que hay errores que son percibidos más fácilmente por otras personas que por los creadores. Las revisiones se clasifican en informales y formales. Las primeras se realizan de forma constante, sin ellas la programación y comprensión de un proyecto sería prácticamente imposible. Por su parte, las revisiones formales constan de 3 elementos: informe escrito del estado del producto revisado,

participación activa y abierta de todos los del grupo de revisión y total responsabilidad de todos los participantes en la calidad de la revisión [25].

Herramientas

Las actividades relacionadas con la gestión de requisitos, anteriormente eran realizadas de forma manual, mediante el uso de procesadores de texto. Un número creciente de herramientas automatizadas, algunas de uso libre y otras comerciales, han surgido para ayudar a definir y aplicar un PDSW efectivo. Con la aparición de modernas herramientas para la definición y gestión de los requisitos se ha logrado un gran avance. Dentro de las herramientas *Computer Aided Software Engineering (CASE)* que sirven de apoyo a los procesos de ingeniería de software, están las especializadas en la administración de requisitos, las mismas se concentran en capturar, administrar y producir una especificación de requisitos.

Las ventajas que proporcionan las herramientas automatizadas para la IR son:

- Permiten un mayor control de proyectos complejos.
- Permiten reducir costos y retrasos en la liberación de un proyecto.
- Permiten una mayor comunicación en equipos de trabajo.
- Ayudan a determinar la complejidad del proyecto y esfuerzos necesarios.

A pesar del desarrollo que presentan las herramientas CASE, existe un número reducido de éstas destinadas a los sistemas GNU/Linux. A continuación se mencionan algunos ejemplos:

Requirements Management (REQMAN)

REQMAN es una herramienta de software libre, bajo licencia *General Public License (GPL)*, solución estructurada de gestión de proyectos genéricos donde se guardan los datos en un lugar central, en el que todos los cambios son trazables hasta el más mínimo detalle. No solo reduce la cantidad de trabajo necesario, sino que también reduce significativamente el riesgo y mejora el resultado final de los proyectos. Es una aplicación web, para el manejo de requisitos y el ciclo de vida de un proyecto, gratis, multi-usuario y que admite la extracción de los datos a Excel y Word [26].

REQMAN incluye gestión de proyectos, gestión de requisitos, registro de tiempo, seguimiento de asuntos, requisito encuentro, la gestión de bases y solicitud de propuesta, aprovisionamiento y solicitud de presupuesto y de la declaración del proveedor.

Ventajas que proporciona REQMAN:

- La reducción del plazo del mercado.
- Mayor rendimiento del proyecto.
- Mejor y más rápida la comunicación del equipo.
- Reducción del costo.
- Objetivos definitivos.
- Claro, no redundante.
- Reducción del tiempo dedicado al trabajo de proyectos administrativos.
- Mejora de los beneficios generales de los proyectos.
- Ofrece tutoriales gratuitos para el uso de la aplicación.

Es una plataforma totalmente integrada al ciclo de vida del proyecto. Ofrece todas las funcionalidades necesarias para administrar sus proyectos de principio a fin, a través de los siguientes módulos funcionales: Planeación y tiempo, Requisitos, Cuestionarios, Reportes, Invitación a propuestas, Flujo de trabajo, Usuarios, Wiki.

Open Source Requirement Management Tool (OSRMT)

Es una herramienta de software libre, bajo licencia GPL, escrita en Java y desarrollada actualmente por Aron Smith y Paul Spencer. Se trata de una herramienta de gestión de requisitos, que permite la descripción avanzada de diversos tipos de requisitos y garantiza la trazabilidad entre todos los documentos relacionados con la IR.

Dentro de las herramientas *open source* que abarcan la IR, se trata sin dudas de una de las que mejores funcionalidades ofrece. Con muchas opciones de configuración, que permite personalizar la herramienta a las necesidades concretas de una organización, esta aplicación cubre los principales aspectos relacionados con la gestión de requisitos: registro, definición, categorización, seguimiento y trazabilidad con el resto de documentos de trabajo.

Ingeniería de Requisitos para Xerberos
CAPÍTULO 1: CONCEPTOS BÁSICOS PARA LA INVESTIGACIÓN

La herramienta ofrece las siguientes funcionalidades: Gestión de requisitos, trazabilidad y representación jerárquica entre todos los documentos de trabajo, personalización y configuración, almacenamiento de ficheros binarios adjuntos o hipervínculos, gestión de usuarios, búsquedas avanzadas (filtros, órdenes) sobre los documentos de trabajo registrados, informes y estadísticos(básicos, específicos creados por el usuario, a partir de los resultados de búsquedas avanzadas, exportados a *HyperText Markup Language* (HTML) o *Portable Document Format* (PDF), exportar información a *Extensible Markup Language* (XML) y HTML, herramienta de migración para los diversos cambios de versiones, múltiples idiomas.

La herramienta posee las siguientes ventajas:

- La visualización de requisitos en forma jerárquica es intuitiva y fácil de manejar.
- Existen diversas distribuciones tanto para un equipo local como para un servidor de aplicaciones permitiendo el desarrollo colaborativo.
- Su licencia es GPL.
- Es un desarrollo en Java por lo que es multiplataforma.
- Las nuevas versiones incorporan un cliente web para permitir accesos desde internet.
- Como herramienta *open source* de gestión de requisitos no tiene muchas competencias en cuanto a las funcionalidades que ofrece.
- Presenta buena documentación.
- El ritmo de mejoras y nuevas versiones es constante.
- Presenta variadas opciones para configurar y personalizar la herramienta a las necesidades concretas de una organización.
- Lleva incorporado un sistema de gestión de la configuración que permite definir líneas base.
- Presenta un gran soporte para mantener la trazabilidad entre los documentos [\[27\]](#).

Luego de realizarse un análisis completo sobre las tendencias de la IR internacionalmente, en Cuba, en la UCI y específicamente en Xerberos, se recomienda para la gestión de requisitos la herramienta

OSRMT. De las herramientas libres presentadas es la que más funcionalidades presenta, con la misma existe facilidad en el manejo de la información en varios formatos (HTML, XML, PDF), también posee una interfaz de escritorio así como una interfaz web facilitando el trabajo de acuerdo con el entorno. Brinda además la posibilidad de personalizar la herramienta de acuerdo con las necesidades concretas del proyecto, presenta buena documentación, su mejoras y nuevas versiones son constantes y fácil de obtener por la funcionalidad que posee de migración.

Xerberos.

Xerberos es un sistema en proceso de desarrollo que permite detectar amenazas de seguridad como: uso indebido de los recursos de la red, acceso ilegal a estaciones o terminales, daños maliciosos y fuga de información sensible. Integra funcionalidades para auditar acciones, procesos y recursos dentro de la empresa brindando notificaciones de posibles modificaciones y actividades indebidas en los mismos. También permite realizar tareas de soporte y mantenimiento en estaciones de trabajo gracias a un sistema de clonación y administración remota de imágenes de sistemas operativos permitiendo gestionar de forma centralizada las particiones, las configuraciones, instalación y desinstalación de software en cada estación. Su interfaz amigable y sencilla permite un fácil acceso a cada funcionalidad del sistema y a los reportes. Está compuesto por varios módulos: Log, Base de datos, Comunicación, Clonación, Monitoreo y Control, Auditorías de Recursos y el de Reportes [28].

En Xerberos, por decisión del equipo de trabajo, el desarrollo estará guiado por la metodología ágil de desarrollo SXP, híbrido cubano de metodologías ágiles creado en la UCI, la cual ofrece una estrategia tecnológica a partir de la introducción de procedimientos ágiles que permiten actualizar los procesos de software para el mejoramiento de la actividad productiva. Se selecciona esta metodología porque fomenta el desarrollo de la creatividad, aumenta el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo. Además, está especialmente indicada para proyectos de pequeños equipos de trabajo. Es orientada a una entrega rápida de resultados y una alta flexibilidad. Ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo seguir de forma clara el avance de las tareas a realizar [29]. Consta de 4 fases principales:

Ingeniería de Requisitos para Xerberos
CAPÍTULO 1: CONCEPTOS BÁSICOS PARA LA INVESTIGACIÓN

- **Planificación-Definición:** Es donde se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
- **Desarrollo:** En esta fase se realiza la implementación del sistema hasta que esté listo para ser entregado.
- **Entrega:** Se entrega el producto final.
- **Mantenimiento:** Esta fase es donde se realiza el soporte para el cliente.

Las fases están conformadas por diferentes flujos de trabajo.

- **Planificación-Definición:** Concepción inicial, Captura de requisitos, Diseño de metáforas.
- **Desarrollo:** Implementación, Pruebas.
- **Entrega:** Entrega de la documentación.
- **Mantenimiento:** Soporte.

En los flujos de trabajo mencionados anteriormente se realizan numerosas actividades. En este proceso solo se tendrán en cuenta las dedicadas a la IR, es decir, dentro de la fase Planificación-Definición en el flujo de trabajo Captura de Requisitos se realizan actividades como: creación de la Lista de Reserva del Producto (LRP), priorización de la LRP, definir las Historias de Usuario, asignar las Historias de Usuario. Generándose de estas actividades los siguientes artefactos: la plantilla LRP y la plantilla Historia de Usuario [\[30\]](#).

Resumiendo

Como resultado de la investigación y del análisis bibliográfico realizado a lo largo de este capítulo, han sido expuestos los principales puntos de interés relacionados con la IR, además, se trataron los aspectos relacionados con el objeto de la investigación y sus principales tendencias. Sobre las bases del marco teórico se establece como propuesta, respondiendo a los objetivos del presente trabajo, la obtención de un conjunto de procesos de IR que garantice con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes. De esta manera, se pretende

minimizar los problemas relacionados por la mala gestión de requisitos en el desarrollo del sistema Xerberos.

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

En este capítulo se presentan las actividades a seguir para el PDSW de Xerberos, resaltando en las mismas las que contemplan la IR. Además, se muestra la propuesta de solución a poner en práctica, definiendo los Procesos de Ingeniería de Requisitos (PIR) para el desarrollo del sistema a partir: de las investigaciones realizadas en sistemas similares a Xerberos, de los procesos de IR definidos por autores como Pressman, Sommerville y Young, de la información acumulada por la experiencia de los especialistas y de las características de Xerberos como producto. Se exponen también las descripciones de cada proceso y el orden de ejecución de los mismos por las fases de desarrollo. Finalmente, se define un cronograma de los procesos, que incluyen las actividades para llevarlos a cabo.

PDSW de Xerberos.

Luego de analizarse en el capítulo anterior las fases y los flujos de trabajo de la metodología de desarrollo SXP e identificar los flujos de trabajo con sus respectivas actividades dedicadas a la IR. Se presentan a continuación las actividades a realizar en el PDSW de Xerberos, para obtener el producto final delimitadas por las fases de SXP y resaltando las que contemplan la IR.

- **Planificación-Definición:**
 - ***Obtener los requisitos para la construcción del sistema.***
 - ***Identificar aplicaciones que brinden soluciones libres para el sistema.***
 - Elaborar propuesta de framework de desarrollo.
 - Diseñar arquitectura.
 - Elaborar propuesta de diseño gráfico.
 - Diseñar la base de datos.
 - ***Concebir las historias de usuario principales.***

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

- Diseñar las clases principales del sistema.

- **Desarrollo:**
 - **Desarrollar el módulo de log.**
 - **Desarrollar el módulo de base de datos.**
 - **Desarrollar el módulo de clonación.**
 - **Desarrollar el módulo de monitoreo y control.**
 - **Desarrollar el módulo de comunicación.**
 - **Desarrollar el módulo de auditoría de recursos.**
 - **Desarrollar el módulo de reportes.**
 - **Desarrollar la interfaz de usuario para administrar el sistema.**
 - Ejecutar y refinar las pruebas.
 - **Desplegar piloto en un entorno controlado.**
 - **Probar la solución integral.**

- **Entrega:**
 - **Corregir errores detectados en el piloto.**
 - Instalar y desplegar del software.

- **Mantenimiento:**
 - Capacitar a usuarios finales.
 - Realizar transferencia tecnológica.

La IR define procesos para crear y mantener los requisitos a lo largo del ciclo de vida del software. Existen diferentes modelos de procesos de IR a seguir según diversos autores. De acuerdo con los procesos que propone Roger S. Pressman, Ian Sommerville, Ralph R. Young, abordados en el

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

capítulo anterior, los cuales pueden ser adoptados dependiendo de las características del proyecto y del producto a desarrollar, para la presente investigación se definen los siguientes: **Inicio del Proyecto** solo es propuesto por Pressman pero se considera importante destinar un proceso a la definición y comprensión del ámbito y la naturaleza del problema a resolver. **Obtención de Requisitos** porque es imprescindible un proceso encargado de obtener las necesidades en forma de requisitos, tanto funcionales como no funcionales del software. **Análisis de Requisitos** porque los requisitos declarados deben ser refinados y validados con el objetivo de clarificar las necesidades y expectativas reales del sistema. **Especificación de Requisitos** porque debe existir el importante proceso de describir correctamente las funciones y desempeño del sistema, además, de las restricciones que van a regir su desarrollo. **Verificación de Requisitos** porque es el proceso que garantiza que se cumplan los requisitos. **Validación de Requisitos** porque confirma que los verdaderos requisitos se aplican en el sistema de entrega. **Gestión de Requisitos** porque los cambios en los requisitos, en el proceso de desarrollo son inevitables y es muy importante controlarlos y darles seguimiento durante el ciclo de vida del proyecto.

A continuación se muestran los PIR definidos anteriormente (**Ver Figura 1**), los mismos se encuentran ubicados en la fase del PDSW de Xerberos donde se ejecutan.

PIR en el desarrollo de Xerberos.



Figura 1: PIR en correspondencia con las fases del PDSW de Xerberos.

Descripción de los PIR propuestos.

En este epígrafe se muestran los procesos por orden de ejecución, definiendo para cada uno de ellos las entradas necesarias para ejecutarlos, su respectiva descripción, los roles que llevan a cabo la acción, las salidas a obtener por cada proceso en cuanto a lo que propone SXP y al programa de mejora que se realiza en estos momentos en la UCI, para alcanzar el nivel 2 de CMMI en todos los centros de producción, así como las técnicas a utilizar. Teniendo en cuenta que Xerberos es un producto atípico, que no presenta cliente y es el clásico tipo de software que se desarrolla en el centro CESOL, donde se obtienen los requisitos y se conforma la concepción del sistema, dependiendo de estudios realizados a sistemas similares e investigaciones.

Inicio del proyecto.

Inicio del proyecto.	
Entradas	<ul style="list-style-type: none">➤ Informes de investigaciones relacionadas con sistemas similares a Xerberos.➤ Descripciones de sistemas similares desarrollados por otras organizaciones u empresas.
Descripción	Es donde se define el ámbito y la naturaleza del problema que se quiere resolver. El objetivo es establecer una comprensión básica del problema, las personas que quieren una solución y la naturaleza de la solución que se desea.
Actividades	<ul style="list-style-type: none">➤ Designar a un miembro del equipo de trabajo encargado de estudiar e investigar acerca de sistemas similares a Xerberos en el mundo.➤ Revisar la información relacionada.➤ Elaborar un informe acerca de las funcionalidades y procesos por los cuales estará compuesto el sistema a desarrollar.➤ Definir involucrados en el proyecto.

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	<ul style="list-style-type: none">➤ Definir la gestión de los requisitos.➤ Elaborar el listado inicial de requisitos sobre la base de la información obtenida a partir de investigaciones y estudios del tema.➤ Elaborar la primera versión de la concepción del sistema.
Roles	Líder de proyecto, analista de sistema.
Salidas	Según lo definido en el programa de mejora Según lo definido en la metodología SXP
	<ul style="list-style-type: none">➤ Modelo conceptual.➤ Modelo de dominio.➤ Especificación de requisitos del software (versión inicial).➤ Criterios para definir proveedores válidos de requisitos.➤ Matriz de disponibilidad.
	<ul style="list-style-type: none">➤ Plantilla de concepción del sistema.➤ Plantilla del modelo de historias de usuario del negocio.➤ Versión inicial de la Plantilla LRP (Lista de Reserva del Producto).
Técnicas	Tormenta de ideas, entrevistas y cuestionarios.

Tabla 1: Proceso inicio del proyecto.

Obtención de requisitos para la construcción del sistema.

En Xerberos, el proceso de obtención de requisitos se realiza mediante varias actividades. Pero la más importante es la identificación de aplicaciones que brinden soluciones libres para el sistema donde también se definen requisitos imprescindibles (**Ver Figura 2**).

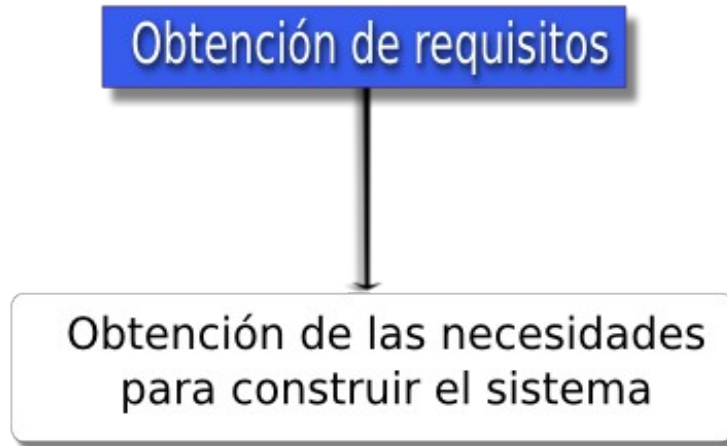


Figura 2: Proceso Obtención de requisitos.

Obtención de las necesidades para construir el sistema.	
Entrada	Listado inicial de requisitos.
Descripción	Es uno de los procesos más importantes y complejos de la disciplina en cuestión. Es este el momento de definir las necesidades, establecer cuáles son los objetivos para el sistema y el producto, qué es lo que se debe lograr, de qué forma el producto satisface las necesidades y cómo se utilizará. Esta información será obtenida a partir del individuo designado para realizar un estudio acerca de los sistemas similares a Xerberos en el proceso Inicio del proyecto.
Actividades	<ul style="list-style-type: none">➤ Preparar condiciones iniciales para las actividades de obtención de requisitos.➤ Realizar la obtención de requisitos y utilizar para ello las técnicas de captura de requisitos.<ul style="list-style-type: none">➤ Analizar los informes y las investigaciones realizadas, y conformar a partir de ellos el listado de requisitos.➤ Identificar aplicaciones que brinden soluciones libres para el

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	sistema.
	<ul style="list-style-type: none">➤ Refinar listado de requisitos funcionales.➤ Refinar listado de requisitos no funcionales.➤ Refinar el glosario de términos.➤ Elaborar Documento de Especificación de Requisitos.➤ Refinar la concepción del sistema de acorde a todo lo definido anteriormente.
Roles	Analista de sistema, cliente (opcional) y usuarios finales.
Salidas	<ul style="list-style-type: none">➤ Según lo definido en el programa de mejora.➤ Glosario de términos.➤ Especificación de requisitos del software.➤ Según lo definido en la metodología SXP.➤ Plantilla de Glosario de Términos.➤ Lista de reserva del producto (LRP).
Técnicas:	Prototipos y análisis jerárquico.

Tabla 2: Proceso obtención de requisitos.

Análisis de requisitos.

El análisis de los requisitos se manifiesta en Xerberos además de ejecutarse como un proceso independiente, en la concepción de las historias de usuario, en el desarrollo de la interfaz de administración y en el desarrollo del sistema (**Ver Figura 3**).

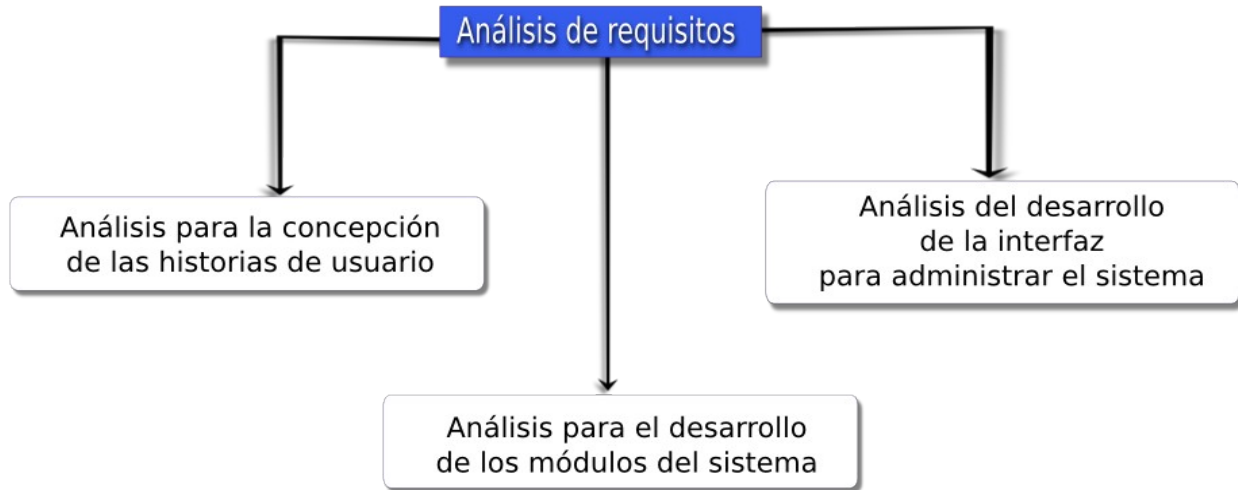


Figura 3: Análisis de requisitos.

Análisis para la concepción de las historias de usuario¹.

Análisis para la concepción de las historias de usuario.	
Entrada	Listado de requisitos.
Descripción	La historia de usuario es la técnica para describir los requisitos del software, sin necesidad de documentaciones extensas. Si son correctamente definidas, guían el proceso de desarrollo. En este proceso se determinan las historias de usuario que conformarán el sistema de acuerdo con los requisitos capturados anteriormente.
Actividades	<ul style="list-style-type: none">➤ Analizar listado de requisitos para determinar cuáles historias de usuario se van a realizar, teniendo en cuenta que varios requisitos pueden formar parte una misma historia de usuario.➤ Asignar a cada historia de usuario para su realización: número, nombre, programador responsable de su implementación, iteración donde se

¹ Existen varias técnicas para describir requisitos: las historias de usuario, los casos de uso, entre otras. En este caso se utiliza la historia de usuario, por ello las actividades en este proceso están orientadas a esa técnica.

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	implementará, prioridad según su importancia (Alta, Media, Baja), puntos estimados (tiempo estimado en semanas) que se le asignará para su implementación, riesgo en desarrollo según programadores (Alto, Medio, Bajo) y puntos reales (tiempo real en semanas).
	<ul style="list-style-type: none">➤ Realizar una breve descripción del proceso que define la historia de usuario.➤ Establecer prototipo de interfaz de todas las interfaces relacionadas con la historia de usuario.
Roles	Analista del Sistema y arquitecto.
Salidas	Según lo definido en el programa de mejora. Según lo definido en la metodología SXP. <ul style="list-style-type: none">➤ Especificación de casos de uso.➤ Evaluación de casos de uso.➤ Especificación de requisitos del software.➤ Plantillas de historias de usuario.
Técnica	Historia de usuario.

Tabla 3: Análisis para la concepción de las historias de usuario.

Análisis para el desarrollo de los módulos del sistema.

Para el desarrollo de Xerberos se han definido diferentes subsistemas o módulos.

Análisis para el desarrollo de los módulos del sistema.	
Entrada	Listado de requisitos.
Descripción	Xerberos está compuesto por varios módulos entre los cuales se encuentra: el módulo de Reportes (encargado del diseño y elaboración de listados o reportes basados en las distintas tablas de la base de datos), el módulo de Clonación (permite el mantenimiento y la administración remota de clientes en

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

cuanto a hardware y software. Su principal objetivo es la clonación de sistemas operativos GNU/Linux en diferentes nodos o estaciones clientes), **el módulo de Comunicación** (Eje central en la comunicación: envío y recepción de mensajes, entre el(los) programa(s) servidor, sus módulos y los clientes), **el módulo de Monitoreo y Control** (encargado de la gestión de seguridad de la información interna. Permite detectar amenazas de seguridad como uso indebido de los recursos de la red, acceso ilegal a estaciones o terminales, daños maliciosos y fuga de información sensible. Además, permite el monitoreo constante sobre el uso de aplicaciones y dispositivos externos de almacenamiento), **módulo de Auditoría de Recursos** (encargado de la auditoría de seguridad de los activos, que se implementa recopilando información sobre los activos de software y hardware), **el módulo de Logs** (permite almacenar información en archivos de texto tanto en el servidor como en los clientes), **el módulo de Base de datos** (constituye un puente o nexo entre la aplicación de administración central (Desktop y Web), los servicios, y las bases de datos que los mismos utilizan). Siendo estos dos últimos módulos utilizados por los restantes, el de log para hacer depuraciones o dar información de salida y el de base de datos para recuperar información o actualizar información en las bases de datos. En este proceso se definen y analizan los requisitos que van a formar parte de los módulos del sistema.

Actividades

- Definir los requisitos funcionales y no funcionales para desarrollar el módulo correspondiente.
- Incorporar los requisitos capturados a la LRP o a la especificación de requisitos.
- ²Diseñar las historias de usuario correspondientes a partir de los requisitos.
- ²Asignar la historia de usuario al programador correspondiente para su

2 En caso de utilizar la técnica historia de usuario.

	desarrollo.
	<ul style="list-style-type: none">➤ Verificar las funcionalidades desarrolladas.➤ Validar las funcionalidades del módulo.➤ Integrar al sistema Xerberos.
Roles	Programador y analista de sistema.
Salidas	Según lo definido en el programa de mejora. Según lo definido en la metodología SXP. <ul style="list-style-type: none">➤ Módulos del sistema.➤ Actualización de la especificación de requisitos del software.
Técnicas	Análisis Jerárquico.

Tabla 4: Análisis para el desarrollo de los módulos del sistema.

Análisis del desarrollo de la interfaz para administrar el sistema.

Análisis del desarrollo de la interfaz para administrar el sistema.	
Entrada	Listado de requisitos.
Descripción	Interfaz gráfica para administrar centralmente Xerberos, fácil de utilizar, amigable e intuitiva para el usuario final. En este proceso se analizan los requisitos para el desarrollo de la misma.
Actividades	<ul style="list-style-type: none">➤ Capturar requisitos funcionales y no funcionales.➤ Precisar las pautas de diseño necesarias para lograr un diseño acorde.➤ Diseñar la(s) funcionalidad(es) correspondientes para lograr la interfaz adecuada.

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	<ul style="list-style-type: none"> ➤ Poner en práctica el proceso de gestión de requisitos para lograr desde un inicio la trazabilidad de los mismos. ➤ Verificar las funcionalidades de la misma. ➤ Validar las funcionalidades.
Roles	Programador y analista de sistema.
Salidas	Según lo definido en el programa de mejora. Según lo definido en la metodología SXP.
	<ul style="list-style-type: none"> ➤ Interfaz gráfica para administrar el sistema. ➤ Actualización de la especificación de requisitos del software.
Técnicas	Prototipos.

Tabla 5: Análisis del desarrollo de la interfaz para administrar el sistema.

Análisis de requisitos.

Análisis de requisitos.	
Entrada	Listado de requisitos.
Descripción	El análisis de requisitos es el proceso encargado de llevar a un nivel superior de detalle los mismos. Convierte los requisitos obtenidos hasta el momento en requisitos reales.
Actividades	<ul style="list-style-type: none"> ➤ Asegurar que los requisitos reflejan las necesidades reales y son correctos. ➤ Documentar cada requisito acorde a los siguientes atributos: <ul style="list-style-type: none"> ➤ Fuente: Involucrado que originó el requisito. ➤ Estado: Propuesto, aceptado o rechazado.

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	<ul style="list-style-type: none">➤ Prioridad: Alto (un requisito de misión crítica, necesaria para la próxima versión), Medio (apoya las operaciones necesarias del sistema, se requiere tiempo, pero podría esperar hasta una versión posterior si es necesario), Bajo (una mejora funcional o de calidad, sería bueno tener algún día, si los recursos lo permiten).➤ Riesgo: Bajo (< 10%), Medio (10 - 50 %) o Alto (> 50 %).➤ Impacto en la arquitectura: Ninguno, extensión o modificación.➤ Estabilidad: % de cambio esperado; Bajo (< 10 %), Medio (10 - 50 %) o Alto (> 50 %).➤ Nivel de peligro/Criticidad: Nula (no hay daño personal o material), baja (puede ser controlada sin lesión personal o significativos daños al sistema), crítica (puede provocar lesiones personales o grandes daños al sistema o requerir acciones correctivas inmediatas para supervivencia de personas y sistemas), catastrófica (puede causar serias lesiones o muerte o pérdida completa del sistema).
Roles	Analista de sistema.
Salidas	Según lo definido en el programa de mejora. Según lo definido en la metodología SXP. <ul style="list-style-type: none">➤ Primera versión de la Descripción de Requisitos.➤ Requisitos rechazados.➤ Evaluación de requisitos.➤ Lista de reserva del producto (LRP) actualizada.
Técnicas	Análisis Jerárquico.

Tabla 6: Análisis de requisitos.

Especificación de requisitos.

Especificación de requisitos.	
Entrada	Listado de requisitos. Concepción del sistema.
Descripción	Da cumplimiento al objetivo de la IR que establece la necesidad de proporcionar los artefactos que permitan crear una profunda comprensión de los requisitos, en función del desarrollo del sistema. Por ello se describe en este proceso cómo garantizar la correcta descripción de los requisitos.
Actividades	<ul style="list-style-type: none">➤ Refinar y revisar los modelos obtenidos.➤ Evaluar calidad, integridad y consistencia de los requisitos. Se propone evaluar la calidad de los requisitos en este proceso usando los criterios establecidos para escribir buenos requisitos:<ul style="list-style-type: none">➤ Necesario: Un requisito es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.➤ Conciso: Un requisito es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.➤ Completo: Un requisito está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.➤ Consistente: Un requisito es consistente si no es contradictorio con otro requisito.➤ No ambiguo: Un requisito no es ambiguo cuando tiene una sola

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
	➤ Verificable: Un requisito es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas.
	➤ Revisar los modelos con los involucrados.
Roles	Analista de sistema.
Salidas	Según lo definido en el programa de mejora. Según lo definido en la metodología SXP.
	➤ Descripción de requisitos. ➤ LRP.
Técnicas	Historia de usuario o casos de uso o descripción de los requisitos.

Tabla 7: Especificación de requisitos.

Verificación de requisitos.

El proceso de verificación se pone de manifiesto al verificar las pruebas y despliegue de solución integral (Ver Figura 4).



Figura 4: Verificación de requisitos.

Verificación de requisitos.					
Entrada	Listado de requisitos.				
Descripción	Consiste en garantizar que se cumplan los requisitos en el funcionamiento de la aplicación. Para ello se prueba y despliega la solución integral del sistema.				
Actividades	<ul style="list-style-type: none"> ➤ Determinar los criterios para comprobar el correcto de funcionamiento de los requisitos. <ul style="list-style-type: none"> ➤ Elaborar listas de chequeo para ello. ➤ Documentar las no conformidades. ➤ Realizar correcciones pertinentes al sistema. ➤ Dar seguimiento a las acciones correctivas. ➤ Modificar el sistema. 				
Roles	Analista de sistema y líder del proyecto.				
Salidas	<table border="0" style="width: 100%;"> <tr> <td style="width: 50%;">Según lo definido en el programa de mejora.</td> <td style="width: 50%;">Según lo definido en la metodología SXP.</td> </tr> <tr> <td> <ul style="list-style-type: none"> ➤ Lista de chequeo para detectar Inconsistencias en REQM. ➤ Salidas del sistema. </td> <td> <ul style="list-style-type: none"> ➤ La metodología no presenta ningún artefacto que contenga esta información. </td> </tr> </table>	Según lo definido en el programa de mejora.	Según lo definido en la metodología SXP.	<ul style="list-style-type: none"> ➤ Lista de chequeo para detectar Inconsistencias en REQM. ➤ Salidas del sistema. 	<ul style="list-style-type: none"> ➤ La metodología no presenta ningún artefacto que contenga esta información.
Según lo definido en el programa de mejora.	Según lo definido en la metodología SXP.				
<ul style="list-style-type: none"> ➤ Lista de chequeo para detectar Inconsistencias en REQM. ➤ Salidas del sistema. 	<ul style="list-style-type: none"> ➤ La metodología no presenta ningún artefacto que contenga esta información. 				
Técnicas	Revisiones técnicas.				

Tabla 8: Verificación de requisitos.

Validación de requisitos. La validación de los requisitos se evidencia mediante la corrección de errores detectados en el piloto (**Ver Figura 5**).



Figura 5: Validación de requisitos.

Validación de requisitos.	
Entrada	Listado de requisitos.
Descripción	Comprueba que los requisitos del sistema sean los correctos de acuerdo con las necesidades que debe satisfacer el producto, definidas en un inicio por el personal designado. A partir de ello se corrigen los errores detectados en el piloto.
Actividades	<ul style="list-style-type: none">➤ Desplegar el sistema en un entorno controlado.➤ Realizar control sobre los requisitos de la versión desplegada con las comprobaciones que se definan.➤ Comprobar que los requisitos son:<ul style="list-style-type: none">➤ Válidos. El sistema provee las funciones que soportan mejor las necesidades del cliente.➤ Consistentes. No existen conflictos entre los requisitos.➤ Completos. Están incluidas todas las funciones requeridas por

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	<p>el usuario.</p> <ul style="list-style-type: none"> ➤ Reales. Los requisitos pueden ser implementados con el presupuesto y tecnologías disponibles. ➤ Verificables. Los requisitos pueden ser verificados. <ul style="list-style-type: none"> ➤ Documentar las deficiencias en una plantilla de No Conformidades. ➤ Realizar las pruebas pertinentes para determinar las deficiencias del sistema. ➤ Refinar la aplicación en general, a partir de los requisitos capturados al principio y el resultado arrojado por las comprobaciones realizadas.
Roles	Personal externo al equipo de desarrollo y el líder del proyecto.
Salidas	<p>Según lo definido en el programa de mejora.</p> <ul style="list-style-type: none"> ➤ Criterios para validar requisitos del cliente. ➤ Criterios para validar requisitos del producto. ➤ Registro de Revisiones de Inconsistencias. <p>Según lo definido en la metodología SXP.</p> <ul style="list-style-type: none"> ➤ La metodología no presenta ningún artefacto que contenga esta información.
Técnicas	Prototipos o despliegue del sistema en un entorno real.

Tabla 9: Validación de requisitos.

Gestión de requisitos.

Gestión de requisitos.	
Entrada	Listado de requisitos.
Descripción	El proceso de gestionar los requisitos de un sistema involucra un conjunto de actividades que ayudan al equipo de desarrollo a identificar, controlar y rastrear los requisitos y los cambios a estos en cualquier momento mientras se desarrolla

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

el proyecto. La gestión de requisitos es un proceso que se desarrolla a lo largo de toda la vida del producto.

Actividades

- Establecer la línea base de los requisitos para la liberación específica del producto.
- Establecer un mecanismo de control de cambio a los requisitos (**Ver Figura 6**).
- Incorporar estrategia y mecanismo anteriores al Documento de Gestión de Requisitos.
- Establecer una herramienta para la gestión de requisitos (*Open Source Requirement Management Tool (OSRMT)*).
- Probar la herramienta en el entorno de trabajo.
- Establecer la composición del repositorio de requisitos.
- Cargar los requisitos a la herramienta.
- Asignar los atributos y estado a cada requisito (id único, especificación, clasificaciones según criterios establecidos, método de verificación, plan de aceptación, resumen, origen, historial de cambios).
- Elaborar las matrices de seguimiento (se entienden por matrices de seguimiento a las relaciones que se establecen entre los requisitos y uno o más aspectos del sistema o de su ambiente). Se propone establecer matrices de rastreabilidad, a continuación se mencionan algunas de estas:
 - **Características:** Relación de los requisitos con características específicas de la aplicación.

CAPÍTULO 2: PROCESOS DE INGENIERÍA DE REQUISITOS A APLICAR EN XERBEROS.

	<ul style="list-style-type: none"> ➤ Fuente: Relación de los requisitos con su fuente. ➤ Dependencia: Relación entre requisitos. ➤ Subsistemas: Relación de los requisitos con el subsistema al que pertenece.
	<ul style="list-style-type: none"> ➤ Controlar los cambios a los requisitos. ➤ Mantener actualizado el estado de cada requisito.
Roles	Analista de sistema.
Salidas	Según lo definido en el programa de mejora. Según lo definido en la metodología SXP. <ul style="list-style-type: none"> • Reporte de Trazabilidad. • Plantilla de Gestión de Cambio.
Técnicas	Revisiones técnicas formales.

Tabla 10: Gestión de requisitos.

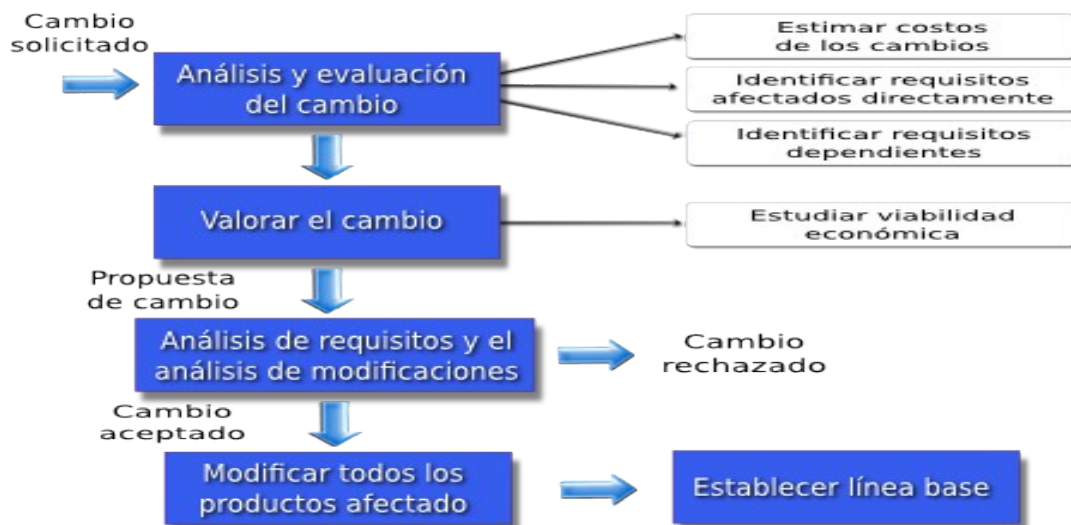


Figura 6: Gestión de cambios en Xerberos.

Cronograma a seguir para implementar los PIR

Luego de definir y describir los PIR, resulta importante destacar que algunos de estos procesos ocurren en paralelo y que todos deben adaptarse a las necesidades del proyecto. Además, están dirigidos a establecer una base sólida respecto al diseño y a la construcción del sistema. Se presenta a continuación (Ver Figura 7) en qué momento debe ejecutarse cada proceso y cómo hacerlo, definiendo cuáles se hacen antes, después o concurrentemente.

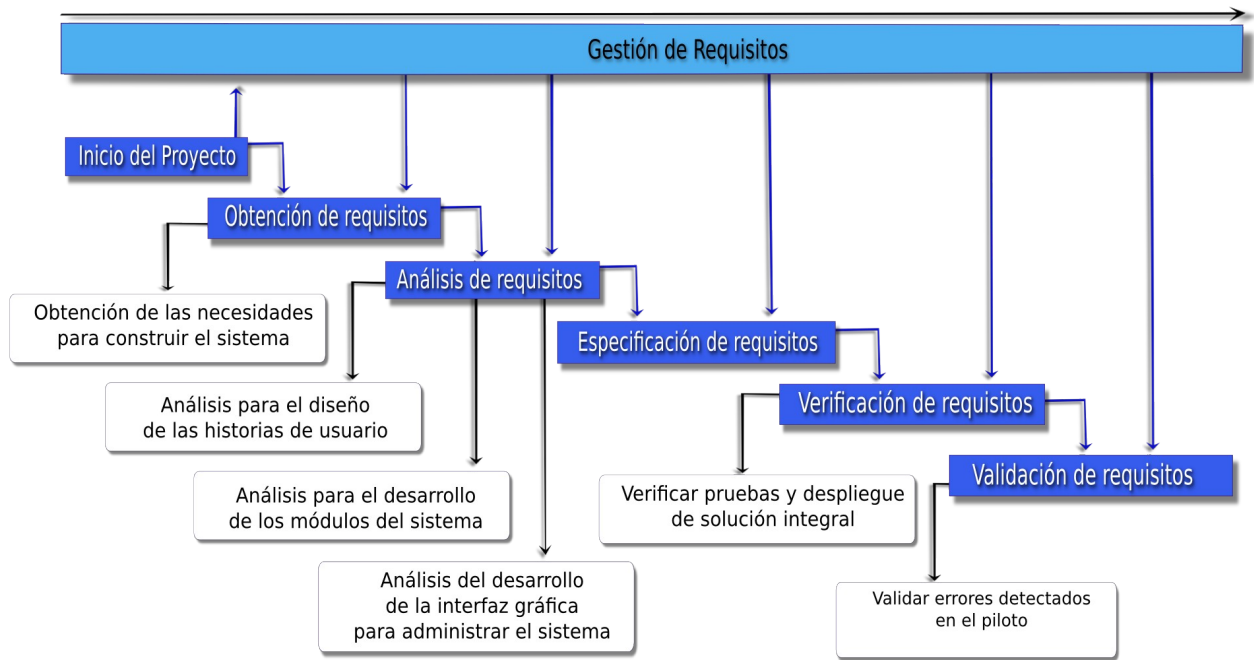


Figura 7: Cronograma a seguir para aplicar los PIR.

En el presente capítulo se obtuvo una propuesta de PIR a seguir para el sistema Xerberos con el objetivo de lograr un producto de calidad y un proceso de IR correctamente definido, puntal importante en el PDSW. Quedando así sentadas las bases para comenzar la evaluación de los PIR en el sistema Xerberos.

CAPÍTULO 3: EVALUACIÓN DE LOS PIR EN EL SISTEMA XERBEROS.

Con el objetivo de evaluar la utilidad y capacidad de los PIR propuestos en el capítulo anterior para Xerberos se traza una estrategia, la misma abarca la utilización del método Delphi, para medir el grado de aceptación de la propuesta entre un grupo de especialistas en el tema.

El panel de expertos

El panel de expertos puede definirse como un grupo de especialistas independientes y reputados en al menos uno de los campos concernidos por el programa que se va a evaluar, al que se reúne para que emita un juicio colectivo y consensuado sobre dicho programa. Según se les solicite, el juicio emitido puede hacer referencia a la puesta en práctica o a los efectos del conjunto o de una parte del programa. Este grupo de trabajo, que se constituye especialmente para la evaluación de acuerdo con una serie de procedimientos estándar, sigue un método de trabajo concreto para celebrar sus reuniones y elaborar su juicio. El método a usar en la presente investigación es la consulta tipo Delphi, la cual se fundamenta en la consulta sistematizada que de forma anónima, se realiza sobre un grupo de especialistas en la materia [\[31\]](#).

Delphi como método de experto.

El método Delphi pretende extraer y maximizar las ventajas que presentan los métodos basados en grupos de expertos y minimizar sus inconvenientes. Para ello se aprovecha la sinergia del debate en el grupo y se eliminan las interacciones sociales indeseables que existen dentro de todo grupo. De esta forma, se espera obtener un consenso lo más fiable posible del grupo de expertos. Este método resulta más efectivo si se garantizan: el anonimato, la retroalimentación controlada y la respuesta estadística de grupo [\[32\]](#).

Características del método Delphi:

- **Anonimato:** No debe existir contacto entre los participantes, pero el administrador/gestor de la encuesta sí puede identificar a cada participante y sus respuestas.

- **Iteración:** Se pueden manejar tantas rondas como sean necesarias.
- **Retroalimentación Controlada:** Los resultados totales de la ronda previa no son entregados a los participantes, solo una parte seleccionada de la información circula.
- **Resultados Estadísticos:** La respuesta del grupo puede ser presentada estadísticamente (promedios y grado de dispersión) [\[33\]](#).

Para la aplicación del método se siguieron las siguientes etapas:

1. Elección de especialistas.
2. Determinación del coeficiente de competencia.
3. Puesta en práctica del método.
 - Elaboración del cuestionario, para la validación de la propuesta.
4. Determinación de la concordancia de los especialistas mediante el coeficiente de Kendall.
5. Análisis de los resultados.

Elección de los especialistas.

Para realizar la elección de especialistas se deben tener en cuenta varios indicadores que garanticen la calidad de los mismos:

La experiencia profesional en el campo que se trate es un requisito indispensable; el experto debe estar muy cualificado en el área objeto de evaluación, y ser reconocido y respetado por sus pares.

La independencia del experto respecto del programa que se va a evaluar es de vital importancia; el evaluador nunca puede ser juez y parte.

La capacidad de trabajo en equipo, para escuchar a los demás, así como una mentalidad abierta, es fundamental. En caso contrario, el clima de trabajo en el panel puede enrarecerse y ello puede desembocar directamente en el fracaso de la misión.

Para la selección existen dos técnicas:

Selección rápida: los responsables de la evaluación tienen acceso a una lista de especialistas reconocidos en un campo determinado y basta con que se aseguren de su independencia respecto del programa evaluado.

Selección por etapas: es conveniente definir primero el perfil de los especialistas de acuerdo con los temas que se vayan a abordar en la evaluación. Para la definición de estos perfiles se tendrán en cuenta los siguientes puntos: el carácter del programa, el grado de controversia (¿los problemas que hay que tratar tienen soluciones alternativas susceptibles de controversia?), los datos disponibles, las incertidumbres (¿contemplan conclusiones del panel?).[\[34\]](#)

En la presente investigación se va a utilizar la técnica de selección rápida. Formaron parte del panel para evaluar la propuesta, 10 posibles especialistas pertenecientes al departamento de ingeniería de software y práctica profesional de la facultad 1, profesores con experiencia en el área de conocimiento pertenecientes al centro CESOL y a otros centros productivos de la UCI, integrantes del equipo de desarrollo de Xerberos y profesionales a cargo del proceso de mejora en la UCI. A cada uno se le evaluaron sus conocimientos y habilidades relacionadas con el PDSW y con la IR.

Determinación del coeficiente de competencia de los especialistas.

[Ver Determinación del coeficiente de competencia de los especialistas.](#)

Luego de confeccionar un listado con las personas que participarán en el cuestionario, se realiza una valoración del nivel de experiencia que poseen sobre la materia. Para ello se realiza una primera pregunta para una autoevaluación de los niveles de argumentación que tienen sobre el tema en cuestión. Marquen con una X, en una escala creciente del 1 al 10, el valor que se corresponde con el grado de conocimiento o información que usted posee sobre la problemática a tratar.

Valor	1	2	3	4	5	6	7	8	9	10
Experto 1									X	
Experto 2									X	
Experto 3										X

Ingeniería de Requisitos para Xerberos
CAPÍTULO 3: EVALUACIÓN DE LOS PIR EN EL SISTEMA XERBEROS.

Experto 4								X		
Experto 5									X	
Experto 6								X		
Experto 7									X	
Experto 8							X			
Experto 9									X	
Experto 10									X	

Tabla 11: Grado de conocimiento o información de los especialistas.

A partir de ello se propone calcular el coeficiente de información o conocimiento (Kc) el cual se obtiene al multiplicar el valor marcado por el experto en la escala de 0 a 10 por 0,1. Se realiza una segunda pregunta que permite valorar un grupo de aspectos que influyen sobre el nivel de argumentación (Ka) del tema a estudiar.

Aquí se determinan los aspectos de mayor influencia. A partir de estos valores reflejados por cada experto en la tabla se contrastan con los valores de una tabla patrón, siendo **Ka = Σ (valores de la tabla patrón por cada nivel marcado en las fuentes de argumentación)**. Una vez obtenido los valores del **Coficiente de Conocimiento (Kc)** y el **Coficiente de Argumentación (Ka)** se procede a obtener el valor del **Coficiente de Competencia (K)** que finalmente es el coeficiente que determina en realidad que participante se toma en consideración para trabajar en la presente investigación. Se calcula de la siguiente forma:

$$K = \frac{Kc + Ka}{2}$$

Especialistas	Fuentes de argumentación						Coficientes		
	F1	F2	F3	F4	F5	F6	Kc	Ka	K
Experto 1	0.2	0.5	0.05	0.05	0.05	0.05	0.9	0,9	0.9

Ingeniería de Requisitos para Xerberos
CAPÍTULO 3: EVALUACIÓN DE LOS PIR EN EL SISTEMA XERBEROS.

Experto 2	0.2	0.5	0.05	0.05	0.05	0.05	0.9	0.9	0.9
Experto 3	0.3	0.5	0.05	0.05	0.05	0.05	1	1	1
Experto 4	0.2	0.4	0.05	0.05	0.05	0.05	0.8	0.8	0.8
Experto 5	0.3	0.4	0.05	0.05	0.05	0.05	0.9	0.9	0.9
Experto 6	0.2	0.5	0.05	0.05	0.05	0.05	0.8	0.9	0.85
Experto 7	0.2	0.5	0.05	0.05	0.05	0.05	0.9	0.9	0.9
Experto 8	0.3	0.4	0.05	0.05	0.05	0.05	0.7	0.9	0.8
Experto 9	0.3	0.5	0.05	0.05	0.05	0.05	0.9	1	0.95
Experto 10	0.3	0.4	0.05	0.05	0.05	0.05	0.85	0.9	0,88

Tabla 12: Fuentes de argumentación de los especialistas.

Posteriormente obtenido los resultados se valoran de la manera siguiente:

$0,8 < K < 1,0$ Coeficiente de Competencia Alto.

$0,5 < K < 0,8$ Coeficiente de Competencia Medio.

$K < 0,5$ Coeficiente de Competencia Bajo.

Se debe utilizar para la consulta a especialistas de competencia alta, no obstante, se puede valorar si utilizar especialistas de competencia media en caso de que el coeficiente de competencia promedio de todos los posibles sea alto, pero nunca se va a utilizar especialistas de competencia baja. Finalmente, se seleccionan los especialistas para participar en la evaluación de la propuesta de IR con el coeficiente de competencia Alto en todos los casos.

Puesta en práctica del método.

El cuestionario elaborado consta de seis preguntas, de las cuales las cinco primeras están encaminadas a evaluar la importancia de la propuesta para la obtención, análisis, especificación y gestión de buenos requisitos, así como su contribución a un desarrollo de software exitoso para Xerberos. Las respuestas a estas preguntas han sido categorizadas en: muy adecuado (MA), bastante adecuado (BA), adecuado (A), poco adecuado (PA) e inadecuado (I). Finalmente, en la

última pregunta se solicita una valoración general sobre la propuesta. El cuestionario fue enviado por correo electrónico a todos los especialistas, explicándoles las características del método y la finalidad del mismo.

Elaboración del cuestionario.

[Ver Cuestionario a especialistas para la evaluación de la propuesta.](#)

Determinación de la concordancia de los especialistas mediante el coeficiente de Kendall.

El **coeficiente de Kendall** mide el grado de asociación entre varios conjuntos (k) de N entidades. Es útil para determinar el grado de acuerdo entre varios jueces, o la asociación entre tres o más variables. El Coeficiente de Concordancia de Kendall (W), ofrece el valor que posibilita decidir el nivel de concordancia entre los especialistas. El valor de W oscila entre 0 y 1. El valor de uno significa una concordancia de acuerdo total y el valor de cero un desacuerdo total. La tendencia a uno es lo deseado pudiéndose realizar nuevas rondas si en la primera no se alcanza significación en la concordancia [35].

A continuación se construye una tabla de Criterios a evaluar/ Especialistas donde se sitúa el rango de valoración que los expertos asignaron a cada uno de los criterios. Estos datos son tomados de los resultados del cuestionario de evaluación (**Ver Tabla 13**).

Criterios	Especialistas										R _j
	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	
C1	3	5	5	5	5	5	5	5	5	5	48
C2	4	4	5	3	5	4	5	3	5	4	42
C3	4	5	4	4	4	5	5	3	3	4	41
C4	5	4	4	4	4	4	4	5	4	4	42
C5	3	5	4	4	5	5	5	3	4	5	43

Tabla 13: Criterios a evaluar contra especialistas

Para comprobar la existencia de concordancia entre los especialistas se definen las siguientes variables:

K - Número de especialistas que intervienen en el proceso de evaluación, por lo que toma el valor de 10.

N - Cantidad de criterios a evaluar. En este caso $N = 5$.

R_j - Suma de los rangos asignados a cada criterio por parte de los especialistas.

\bar{R}_j - Media de los rangos y se determina mediante la fórmula:

$$\bar{R}_j = \frac{\sum_{i \rightarrow j}^N R_j}{N} = \frac{48 + 42 + 41 + 42 + 43}{5} = 43,2$$

S - es la suma de los cuadrados de las desviaciones y se calcula de la siguiente forma:

$$\begin{aligned} S &= \sum_{i \rightarrow j}^N (R_j - \bar{R}_j)^2 = (48 - 43,2)^2 + (42 - 43,2)^2 + (41 - 43,2)^2 + (42 - 43,2)^2 + (43 - 43,2)^2 \\ &= 23,04 + 1,44 + 4,84 + 1,44 + 0,04 = 30,8 \end{aligned}$$

W - es el coeficiente de Kendall y se calcula mediante la fórmula siguiente:

$$W = \frac{12S}{K^2(N^3 - N)} = \frac{12 * 30,8}{10^2 * (5^3 - 5)} = \frac{369,6}{12000} = 0,0308$$

Una vez obtenido W se procede con el cálculo del Chi-Cuadrado para poder comprobar si existe concordancia entre los especialistas, el mismo se obtiene a través de la fórmula siguiente:

$$X^2_{real} = K(N-1)W = 10 * (5-1) * 0,0308 = 1,232$$

Este Chi-Cuadrado real calculado se compara con el de la tabla inversa de la función de distribución de la variable Chi-Cuadrado con una probabilidad de error de 0,01:

Si $K < 7$ se busca S para su valor crítico donde W sea de significación 0.05 y 0.01.

Si $K \geq 7$ se calcula Chi cuadrado real con significación $\alpha = 0.01$ y un nivel de confianza del 99 % [35].

Si $X^2_{real} < X^2(\alpha, N-1)$, entonces existe concordancia en el trabajo de los especialistas. Luego de la realización de los cálculos pertinentes, estos arrojaron que $X^2_{real}=1,232$ y el $X^2(0.01,4)=13,277$ lo cual corrobora el cumplimiento de la comparación y por tanto, existe concordancia entre los especialistas.

Análisis de los resultados.

A continuación se muestran los resultados de los 5 criterios evaluados a modo de resumen por cada especialista (Ver Figura 8 y Figura 9).

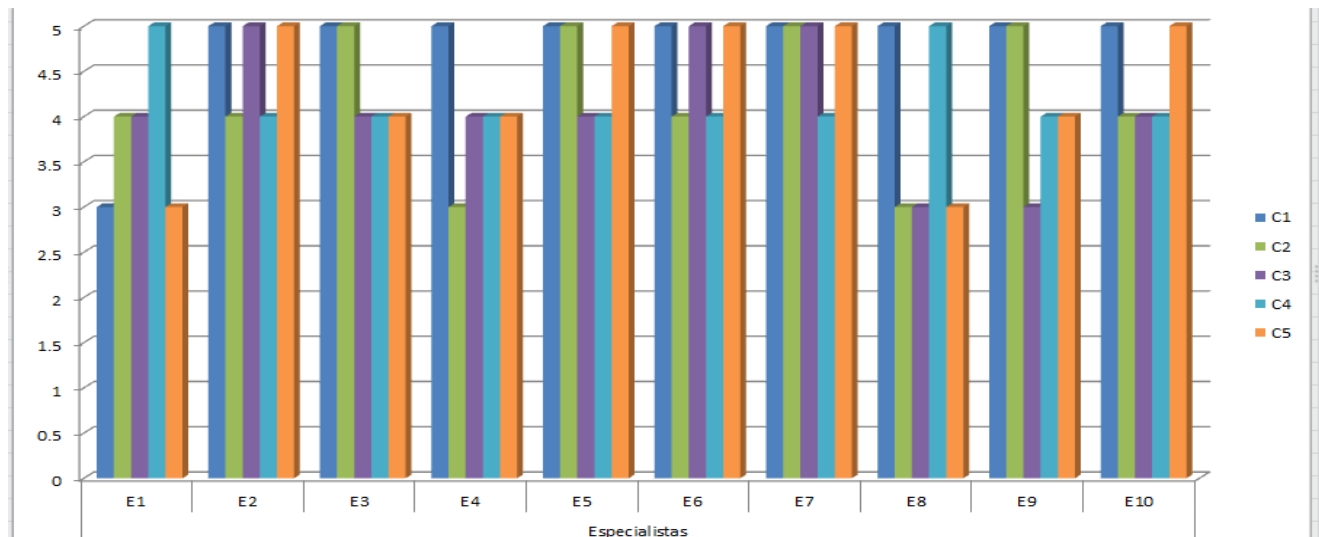


Figura 8: Resumen de evaluaciones.

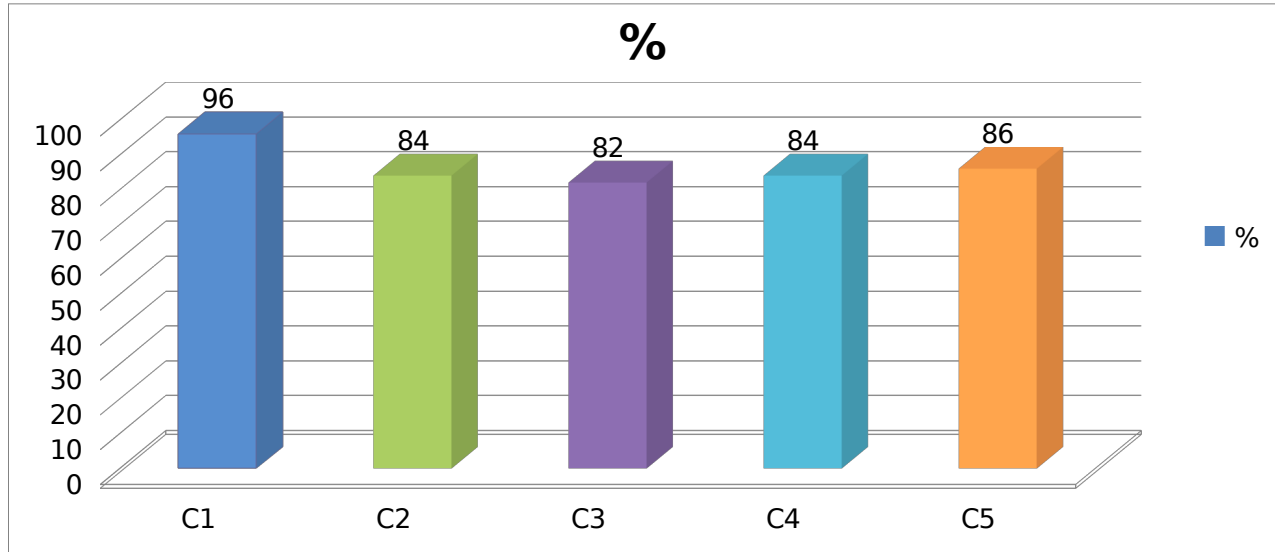


Figura 9: Resumen de evaluaciones en por ciento.

Para llevar a cabo el análisis de los resultados se utilizaron métodos estadísticos tal y como indica el método Delphi. Inicialmente es necesario obtener la tabla de frecuencia absoluta para cada criterio a evaluar de acorde a las cinco categorías que existen: (Muy adecuado (MA), Bastante adecuado (BA), Adecuado (A), Poco Adecuado (PA), Inadecuado (I)) (**Ver Tabla 14**).

No	Criterios	MA	BA	A	PA	I	Total
1	Importancia de la propuesta.	9	0	1	0	0	10
2	Contribución a la obtención de un producto final con calidad.	4	4	2	0	0	10
3	Propuesta correcta y completa.	3	5	2	0	0	10
4	Adecuación de las actividades que componen cada proceso.	2	8	0	0	0	10
5	Evaluación de la propuesta.	5	3	2	0	0	10

Tabla 14: Frecuencias Absolutas.

Ingeniería de Requisitos para Xerberos
CAPÍTULO 3: EVALUACIÓN DE LOS PIR EN EL SISTEMA XERBEROS.

A partir de la frecuencia absoluta se obtienen las frecuencias acumuladas, donde cada número en la fila, excepto el primero se obtiene sumándole el anterior tal y como se muestra (**Ver Tabla 15**).

No	Criterios	MA	BA	A	PA	I
1	Importancia de la propuesta.	9	9	10	10	10
2	Contribución a la obtención de un producto final con calidad.	4	8	10	10	10
3	Propuesta correcta y completa.	3	8	10	10	10
4	Adecuación de las actividades que componen cada proceso.	2	10	10	10	10
5	Evaluación de la propuesta.	5	8	10	10	10

Tabla 15: Frecuencias Acumuladas.

La frecuencia absoluta acumulada se divide entre la cantidad de especialistas (10) y se obtiene la frecuencia relativa acumulada tal y como aparece en (**Ver Tabla 16**).

No	Criterios	MA	BA	A	PA	I
1	Importancia de la propuesta.	0.9	0.9	1	1	1
2	Contribución a la obtención de un producto final con calidad.	0.4	0.8	1	1	1
3	Propuesta correcta y completa.	0.3	0.8	1	1	1
4	Adecuación de las actividades que componen cada proceso.	0.2	1	1	1	1
5	Evaluación de la propuesta.	0.5	0.8	1	1	1

Tabla 16 Frecuencia Relativa Acumulada.

Por último, se buscan las imágenes de las frecuencias relativas acumuladas por medio de la función (Distribución Normal. Estandar Invertida) y se adicionan a la tabla las siguientes salidas:

CAPÍTULO 3: EVALUACIÓN DE LOS PIR EN EL SISTEMA XERBEROS.

- **Suma:** Sumatoria de cada fila y de cada columna según sea el caso.
- **P:** Promedio de la suma de cada fila.
- **N:** División de la sumatoria de las sumas de las filas entre el resultado de multiplicar el número de categorías (5-1 porque se elimina la categoría Inadecuado para el resumen de los resultados) por el número de criterios (5).
- **N-P:** Es el valor promedio que le otorgan los especialistas consultados a cada pregunta.
- **Punto de corte:** Promedio de la suma de cada columna.

La **Tabla 17** resume estos resultados:

							N=0,6316		
N o	Criterios	MA	BA	A	PA	Suma	P	N-P	Categorías
1	Importancia de la propuesta.	0,82	0,82	0,84	0,84	3,32	0,83	-0,1 984	Muy adecuado
2	Contribución a la obtención de un producto final con calidad.	0,65	0,79	0,84	0,84	3,12	0,78	-0,1 484	Muy adecuado
3	Propuesta correcta y completa.	0,62	0,79	0,84	0,84	3,09	0,77	-0,1 384	Muy adecuado
4	Adecuación de las actividades que componen cada proceso.	0,58	0,84	0,84	0,84	3,1	0,78	-0,1 484	Muy adecuado
5	Evaluación de la propuesta.	0,69	0,79	0,84	0,84	3,16	0,79	-0,1 584	Muy adecuado
	Suma	3,36	4,03	4,2	4,2	15,79			
	Puntos de corte	0,67	0,81	0,84	0,84				

Tabla 17: Resumen de resultados

Los puntos de corte se utilizan para determinar la categoría o grado de adecuación de cada criterio según la opinión de los especialistas consultados. Para determinar cuál es el grado de adecuación de cada criterio a evaluar se realiza como muestra la **Tabla 18**.

Muy adecuado	Bastante adecuado	Adecuado	Poco Adecuado	Inadecuado
$N-P \leq 0,67$	$0,67 > N-P \leq 0,81$	$0,81 > N-P \leq 0,84$	$0,84 > N-P \leq 1$	-

Tabla 18: Grado de adecuación

A partir de este análisis, de acuerdo con el criterio de especialistas en el tema la propuesta fue considerada muy adecuada ya que responde a los objetivos trazados en la investigación. La aplicación del método de especialistas permitió evaluar la necesidad e importancia del empleo de los procesos de IR propuestos en Xerberos, así como su contribución a la mejora en el desarrollo del mismo.

CONCLUSIONES

Los resultados obtenidos durante el desarrollo de este trabajo han sido los esperados y de acuerdo con los objetivos que fueron definidos inicialmente. Entre otros aspectos significativos que apoyaron el desarrollo de la presente investigación se pueden destacar:

- El estudio acerca de las herramientas, los procesos, las técnicas, los modelos, estándares y normas que contemplan los requisitos permitieron sentar las bases en la investigación para luego diseñar los procesos adecuados según las características del sistema.
- Los PIR definidos para el desarrollo de Xerberos van a permitir lograr una correcta captura y gestión de los requisitos, puntal importante para lograr un producto final con calidad.
- La aplicación del método de expertos Delphi permitió evaluar la propuesta de IR, comprobando que la misma es correcta y completa. Además, las actividades que la componen son las adecuadas y contribuyen a la obtención de un producto final con calidad.

RECOMENDACIONES

Para futuras investigaciones y proyectos que guarden relación con la presente investigación se recomienda:

- Construir el sistema automatizado Xerberos a partir de la propuesta realizada.
- Aplicar la propuesta presentada en software con procesos similares a Xerberos.

REFERENCIAS BIBLIOGRÁFICAS

1. **M.C.C Hernández Islas, Luis. Ingeniería de Requerimientos - GEZZLER.** [citado 20 octubre 2011]. Available from world wide web: <<http://es.scribd.com/GEZZLER/d/22413797-Ingenieria-de-Requerimientos-GEZZLER>>.
2. **PÉREZ ROLDÁN, Dayron, TORRES SÁNCHEZ, José Ernesto. Breve descripción del sistema Xerberos.** Expediente de proyecto de Nova-Xerberos. Universidad de las Ciencias Informáticas. [citado 25 Octubre 2011]. Available from world wide web: <https://repositorio.geitel.prod.uci.cu/svn/xerberos/Xerberos/stable/descripcion_de_xerberos.odt>
3. **Pakoz, Zuñiga. Procesos De La Ingeniería De Requerimientos.** 2004. [citado 26 Octubre 2011]. Available from world wide web: <<http://www.mitecnologico.com/Main/ProcesosDeLaIngenieriaDeRequerimientos>>.
4. **Pressman, Roger S.** 2006, “Ingeniería del Software: Un enfoque práctico”, Sexta edición, México DF, Editorial McGraw Hill.
5. **Sommerville, Ian.** 2005, “Ingeniería del Software”, Séptima edición, México DF, Editorial Pearson.
6. **Arias Chaves, Michael.** La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software revista InterSedes. VI, 07 2007. Available from world wide web: <<http://www.intersedes.ucr.ac.cr>>.
7. **IEEE Std 830 (1998), IEEE Guide for Developing System Requirements Specifications,** <<http://www.standards.ieee.org> >
8. **Idem [5]**
9. **Herrera J, Lizka Johany. Ingeniería De Requerimientos - Ingeniería De Software - Ilustrados!** [citado 31 Octubre 2011]. Available from world wide web: <<http://www.ilustrados.com/tema/1605/Ingenieria-Requerimientos-Ingenieria-Software.html>>.
10. **Wiegers, Karl E. First Things First: Prioritizing Requirements.** [citado 5 Noviembre 2011]. Disponible en la Web: <<http://www.processimpact.com/articles/prioritizing.html>>.

11. **Valladolid Aguinaga, Javier. Especificación de requisitos software.** June 1999. [citado 11 Noviembre 2011]. Available from world wide web: <<http://www.infor.uva.es/~descuder/proyectos/ipo/requi.htm>>.
12. **González Delgado, Diana Maricela. Pym.** 2009. [citado 12 Noviembre 2011]. Disponible en la Web: <<http://www.slideshare.net/borboletiu/pym>>.
13. **Leffingwell, Dean, Widrig, Don. Managing Software Requirements: A Use Case Approach, Second Edition.** s.l.: Addison Wesley, 2003.
14. **Pressman, Roger S**2006, “Ingeniería del Software: Un enfoque práctico”, Sexta edición, México DF, Editorial McGraw Hill.
15. **Sommerville, Ian** 2005, “Ingeniería del Software”, Séptima edición, México DF, Editorial Pearson.
16. **Young, Ralph R. The Requirements Engineering Handbook.** Norwood: ARTECH HOUSE, INC, 2004.
17. **Ferreira Aranda, Juan Marcelo, Gómez, Silvano Christian, Rodas, Marcelo. Introducción a CMMI.** [citado 19 Noviembre 2011]. Available from world wide web: <<http://www.slideshare.net/albinogoncalves/introduccion-a-cmmi>>.
18. **Centro de calidad para soluciones informáticas, UCI. Página inicio programa de mejoras.** [citado 23 Noviembre 2011]. Available from world wide web: <<http://calisoft.uci.cu/index.php/proceso-de-mejora>>.
19. **Martinez, Evelio. Estándares y organizaciones.** [citado 11 Diciembre 2011]. Available from world wide web: <<http://www.eveliux.com/mx/estandares-y-organizaciones.php>>.
20. **IEEE Std 830 (1998). IEEE Especificaciones de los requisitos del software,** [citado 9 Diciembre 2011] Available from world wide web: <http://www.ctr.unican.es/asignaturas/is1/IEEE830_esp.pdf>
21. **IEEE Std 610.12 (1990). IEEE Standard Glossary of Software Engineering Terminology,** [citado 9 Diciembre 2011]. Available from world wide web: <<http://www.standards.ieee.org>>.
22. **IEEE Std 1012-1998 Standard for Software Verification and Validation.** New York : s.n.,1998.

23. Torres de Paz, Rosa María. **El proceso de Ingeniería de Requisitos en el ciclo global del software.** [citado 11 Diciembre 2011] Available from world wide web: <<http://www.lsi.us.es/docs/doctorado/memorias/memo-inv-rosa-m-torres.pdf>>.
24. Cohn, Mike. **User Stories Applied.** Addison Wesley, 2004.
25. López Quesada, Juan Antonio. **Fundamentos de Ingeniería del Software Departamento de Informática y Sistemas.** [citado 28 Enero 2012]. Disponible en la Web: <<http://www.slidefinder.net/t/tema8calidad/tema8calidad/7853786>>.
26. Silva, Andrés. **Herramientas libres de gestión de requisitos.** [citado 29 Enero 2012]. Available from world wide web: <<http://asilva.wordpress.com/2006/11/01/herramientas-libres-de-gestion-de-requisitos/>>.
27. Nieto Rubio, José. **Promoción del desarrollo de SW libre en un entorno de calidad y confianza adaptando las metodologías, procesos, modelos de negocio y últimas tecnologías.** [citado 9 Febrero 2012]. Available from world wide web: <<http://www.ines.org.es/vulcano/wp-content/uploads/2010/05/d552.pdf>>.
28. Idem [1]
29. Peñalver Romero, Gladys Marsi, García de la Puente, Sergio Jesus, Meneses, Abel Abad. **SXP, metodología de desarrollo de software.** 2010.
30. Peñalver Romero, Gladys Marsi. **MA-GMPR-UR2 Metodología ágil para proyectos de software libre.** Junio 2008.
31. Estrella, Ana matilde. **ITSON | Métodos para Evaluar Necesidades | Método Delphi.** [citado 1 Marzo 2012]. Available from world wide web: <http://biblioteca.itson.mx/oa/ciencias_administrativa/oa8/metodos_necesidades_entorno/p6.htm>.
32. Hurtado de Mendoza Fernández, Sandra. **Criterio de Expertos. Su procedimiento a través del método delphy.** *Histodidáctica.* [citado 1 Marzo 2012]. Available from world wide web: <<http://www.ub.edu/histodidactica/Epistemologia/Delphy.htm>>.
33. Astigarraga, Eneko. **METODO DELPHI.** [citado 5 Marzo 2012] Available from world wide web: <www.prospectiva.eu/zaharra/03_Delphi_ESTE.pdf>.

34. **EuropeAid - Evaluation - Guidelines.** [citado 2 Mayo 2012]. Disponible en la Web:
<http://ec.europa.eu/europeaid/evaluation/methodology/tools/too_pan_how_es.htm>.
35. **Cuesta Santos, Armando. La toma de decisiones consensuales.** Facultad de Ingeniería Industrial UNIVERSIDAD TECNOLÓGICA DE LA HABANA : s.n.

BIBLIOGRAFÍA

1. **Astigarraga, Eneko. METODO DELPHI.** [citado 5 Marzo 2012] Available from world wide web: <www.prospectiva.eu/zaharra/03_Delphi_ESTE.pdf>.
2. **Centro de calidad para soluciones informáticas, UCI. Página inicio programa de mejoras.** [cited 23 November 2011]. Available from world wide web: <<http://calisoft.uci.cu/index.php/proceso-de-mejora>>.
3. **Cohn, Mike. User Stories Applied.** Addison Wesley, 2004.
4. **Cuesta Santos, Armando. La toma de decisiones consensuales.** Facultad de Ingeniería Industrial UNIVERSIDAD TECNOLÓGICA DE LA HABANA : s.n. **Soren Lauensen. Software Requirements-Style and Technicles.** Great Britain by Biddles Ltd, King's Lynn, Norfolk.
5. **Chaves, Michael Arias. La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software revista InterSedes. VI, 07 2007.** Available from world wide web: <<http://www.intersedes.ucr.ac.cr>>.
6. **Durán,A., Bernández, B. “Metodología para la Elicitación de Requisitos de Software”**Universidad de Sevilla.2002.
7. **EuropeAid - Evaluation - Guidelines.** [citado 2 Mayo 2012]. Disponible en la Web: <http://ec.europa.eu/europeaid/evaluation/methodology/tools/too_pan_how_es.htm>.
8. **Escalona, M.J., Koch, N. 2005. Ingeniería de Requisitos en Aplicaciones para la Web: Un estudio comparativo.** Sevilla : s.n., 2005.
9. **Estrella, Ana matilde. ITSON | Métodos para Evaluar Necesidades | Método Delphi.** [citado 1 Marzo 2012]. Available from world wide web: <http://biblioteca.itson.mx/oa/ciencias_administrativa/oa8/metodos_necesidades_entorno/p6.htm>.
10. **F, Ian, Stevens, Alexander Richard. Writing Better Requirements.** Pearson Educación, SA 2002.

11. **FEBLES ESTRADA, Ailyn. ConfigCase, Un modelo de referencia para la Gestión de Configuración en las Pymes.** Tesis de Doctorado. CUJAE. Ciudad de la Habana, 2003.
12. **Febles Estrada, Ailyn. Ejemplo de validación de investigación.** [citado 1 Marzo 2012]. Available from world wide web: <<http://es.scribd.com/doc/16166360/Ejemplo-de-validacion-de-investigacion>>.
13. **Ferreira Aranda, Juan Marcelo, Gómez, Silvano Christian, Rodas, Marcelo. Introducción a CMMI.** [citado 19 Noviembre 2011]. Available from world wide web: <<http://www.slideshare.net/albinogoncalves/introduccion-a-cmmi>>.
14. **González Delgado, Diana Maricela. Pym.** 2009. [citado 12 Noviembre 2011]. Disponible en la Web: <<http://www.slideshare.net/borboletiu/pym>>.
15. **González Ferro, Osmany. Un acercamiento a la trazabilidad en el desarrollo ágil de software.** septiembre 2011.
16. **HERNÁNDEZ AGUILAR, Violena. Protofase a la ingeniería de requisitos para facilitar la comprensión del negocio a informatizar en el desarrollo de software de gestión.** Tesis para optar por el Título académico de Máster en Gestión de Proyectos Informáticos. Universidad de las Ciencias Informáticas. Ciudad de la Habana, Noviembre 2009.
17. **Herrera J, Lizka Johany. Ingeniería De Requerimientos - Ingeniería De Software - Ilustrados!** [citado 31 Octubre 2011]. Available from world wide web: <<http://www.ilustrados.com/tema/1605/Ingenieria-Requerimientos-Ingenieria-Software.html>>.
18. **Hurtado de Mendoza Fernández, Sandra. Criterio de Expertos. Su procedimiento a través del método delphy. *Histodidáctica*.** [citado 1 Marzo 2012]. Available from world wide web: <<http://www.ub.edu/histodidactica/Epistemologia/Delphy.htm>>.
19. **IEEE Std 830 (1998), IEEE Guide for Developing System Requirements Specifications,** <<http://www.standards.ieee.org>>
20. **IEEE Std 610.12 (1990), IEEE Standard Glossary of Software Engineering Terminology,** <<http://www.standards.ieee.org>>.

21. **IEEE Std 830 (1998). IEEE Especificaciones de los requisitos del software.** [citado 9 Diciembre 2011]. Available from world wide web: <http://www.ctr.unican.es/assignaturas/is1/IEEE830_esp.pdf>
22. **IEEE Std 1012-1998 Standard for Software Verification and Validation.** New York : s.n.,1998.
23. **Islas Hernández, M.C.C. Luis. Ingeniería de Requerimientos - GEZZLER.** [citado 20 octubre 2011]. Available from world wide web: <<http://es.scribd.com/GEZZLER/d/22413797-Ingenieria-de-Requerimientos-GEZZLER>>.
24. **Silva, Andrés. Herramientas libres de gestión de requisitos.** [citado 29 Enero 2012]. Available from world wide web: <<http://asilva.wordpress.com/2006/11/01/herramientas-libres-de-gestion-de-requisitos/>>.
25. **Móráguez Iglesias, Arabel. El método Delphi | GestioPolis. May 2006.** [citado 5 Marzo 2012]. Available from world wide web: <<http://www.gestiopolis.com/canales6/eco/metodo-delphi-estadistica-de-investigacion-cientifica.htm>>.
26. **Iglesias, Cecilia. OSRMT: Open Source Requirements Management Tool | IPCorp Blog.** December 2008. [citado 9 Febrero 2012]. Available from world wide web: <<http://www.ipcorp.com.ar/blog/2008/12/11/osrmt-open-source-requirements-management-tool/>>.
27. **Leffingwell, Dean y Widrig, Don. Managing Software Requirements: A Use Case Approach, Second Edition.** s.l.: Addison Wesley, 2003.
28. **Londoño, Luis Fernando,*Anaya, Raquel,**Tabares, Marta Silvia. ANÁLISIS DE LA INGENIERÍA DE REQUISITOS ORIENTADA POR ASPECTOS SEGÚN LA INDUSTRIA DEL SOFTWARE.** Julio 2008.
29. **López Quesada, Juan Antonio. Fundamentos de Ingeniería del Software Departamento de Informática y Sistemas.** [citado 28 abril 2012]. Disponible en la Web: <<http://www.slidefinder.net/t/tema8calidad/tema8calidad/7853786>>.

30. Llechú Ramos, Saily, Torres Torres, Dainet. **Procesos de ingeniería de requisitos en el desarrollo de distribuciones de GNU/Linux.** junio 2011.
31. **MARANTE VALDIVIA, Marbys.** **Proceso para planear la cartera de servicios en la adopción de una iniciativa SOA.** Tesis para optar por el Título académico de Máster en Gestión de Proyectos Informáticos. Universidad de las Ciencias Informáticas. Ciudad de la Habana, Julio 2010.
32. **PÉREZ ROLDÁN, Dayron, TORRES SÁNCHEZ, José Ernesto.** **Breve descripción del sistema Xerberos.** Expediente de proyecto de Nova-Xerberos. Universidad de las Ciencias Informáticas. [citado 25 Octubre 2011]. Available from world wide web: https://repositorio.geitel.prod.uci.cu/svn/xerberos/Xerberos/stable/descripcion_de_xerberos.odt
33. **Página inicio programa de mejoras.** [citado 30 abril 2012]. Disponible en la Web: <http://calisoft.uci.cu/index.php/proceso-de-mejora>.
34. **Pakoz, Zuñiga.** **Procesos De La Ingenieria De Requerimientos.** 2004. [citado 26 Octubre 2011]. Available from world wide web: <http://www.mitecnologico.com/Main/ProcesosDeLaIngenieriaDeRequerimientos>.
35. **Peñalver Romero, Gladys Marsi, García de la Puente, Sergio Jesus, Meneses Abad, Abel.** **SXP, metodología de desarrollo de software.** 2010.
36. **Peñalver Romero, Gladys Marsi.** **MA-GMPR-UR2 Metodología ágil para proyectos de software libre.** Junio 2008.
37. **Pressman, Roger S.** 2006, “**Ingeniería del Software: Un enfoque práctico**”, Sexta edición, México DF, Editorial McGraw Hill.
38. **Pressman, Roger S.** **SOFTWARE ENGINEERING.A PRACTITIONERS APPROACH.** Seventh edition Editorial McGraw Hill.
39. **Project Management Institute (2004),** **Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK),** Tercera Edición, EE.UU, www.pmi.org/Pages/default.aspx
40. **Silva de la Hera, Maggie Arminda.** **Propuesta de un modelo de Ingeniería de Requisitos para los proyectos del Polo Video y Sonido Digital.** Mayo 2009.

41. **Sommerville, Ian** 2005, “**Ingeniería del Software**”, Séptima edición, México DF, Editorial Pearson.
42. **Soto López, Nilet María**. **GESTIÓN DE REQUISITOS APLICADA A PROYECTOS PARA EL DESARROLLO DE SISTEMAS DE GESTIÓN POLICIAL**. Mayo 2010.
43. **Tabares, Marta Silvia, Barrera, Andrés Felipe, Arroyave, Juan David, Pineda, Juan Diego**. **UN MÉTODO PARA LA TRAZABILIDAD DE REQUISITOS EN EL PROCESO UNIFICADO DE DESARROLLO**. Número 8, diciembre 2007.
44. **Torres de Paz, Rosa María**. **El proceso de Ingeniería de Requisitos en el ciclo global del software**. Available from world wide web: <<http://www.lsi.us.es/docs/doctorado/memorias/memo-inv-rosa-m-torres.pdf>>.
45. **Valladolid Aguinaga, Javier**. **Especificación de requisitos software**. June 1999. [citado 11 Noviembre 2011]. Available from world wide web: <<http://www.infor.uva.es/~descuder/proyectos/ipo/requi.htm>>.
46. **Windle, Daniel R, René Abreo, L**. **Software Requirements Using the Unified Process**. [Upper Saddle River, New Jersey 07458]: Prentice Hall PTR.
47. **Wiegars, Karl E**. **First Things First: Prioritizing Requirements**. [citado 5 Noviembre 2011]. Disponible en la Web: <<http://www.processimpact.com/articles/prioritizing.html>>.
48. **Young, Ralph R**. *The Requirements Engineering Handbook*. Norwood: ARTECH HOUSE, INC, 2004.

ANEXO

Anexo1: Determinación del coeficiente de competencia de los especialistas.

Para la evaluación de la investigación Proceso de Ingeniería de Requisitos para Xerberos se desea someter la misma a la valoración de un grupo de especialistas, para ello es necesario conocer sus conocimientos y habilidades en cuanto al proceso de desarrollo de software y a la ingeniería de requisitos.

Responda las siguientes preguntas:

1. Marquen con una X, en una escala creciente del 1 al 10, el valor que se corresponde con el grado de conocimiento o información que tiene sobre la temática que se investiga.

Valor	1	2	3	4	5	6	7	8	9	10

2. Sobre el grado de influencia que cada una de las fuentes de argumentación presentadas a continuación ha tenido en su conocimiento sobre el tema que se investiga, marque con una X.

Fuentes de argumentación	Nivel		
	Alto	Medio	Bajo
Análisis teóricos realizados por usted.			
Su experiencia obtenida.			
Trabajos de autores nacionales.			
Trabajos de autores extranjeros.			
Su conocimiento del estado del problema en el extranjero.			

Su intuición.			
---------------	--	--	--

Anexo2: Cuestionario a especialistas para la evaluación de la propuesta.

Usted forma parte del panel de especialistas seleccionados para la presente investigación: Proceso de Ingeniería de Requisitos (IR) para Xerberos. Debe responder las preguntas en su totalidad, para su conocimiento el presente cuestionario es totalmente anónimo.

En el anexo 1 del presente cuestionario se muestra la propuesta de procesos de IR para Xerberos. Es necesario que luego de su análisis y comprensión emita su criterio, primeramente de forma numérica dándole una calificación del 5 al 1, correspondiendo a cada uno de los valores los siguientes criterios respectivamente: Muy Adecuado, Bastante Adecuado, Adecuado, Poco Adecuado e Inadecuado; y finalmente emita por escrito si lo desea su criterio de forma más amplia, así como sus recomendaciones.

Criterios a Evaluar

- C1: Importancia de la propuesta. C2: Contribución a la obtención de un producto final con calidad.
- C3: Propuesta correcta y completa. C4: Adecuación de las actividades que componen cada proceso.
- C5: Evaluación de la propuesta.

Criterios a Evaluar					
	C1	C2	C3	C4	C5
Calificación					

Opinión: _____

Nota: De parte del equipo de desarrollo le agradecemos su participación e importante contribución para el perfeccionamiento del sistema Xerberos, favor de enviar sus respuestas a la dirección mdelgado@estudiantes.uci.cu .

GLOSARIO DE TÉRMINOS

CASE: Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costos de las mismas en términos de tiempo y de dinero.

CMMI: Integración de Modelos de Madurez de Capacidades o Capability Maturity Model Integration (CMMI) es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software.

Concepción del sistema: En esta plantilla se refleja la visión general del producto a implementar, recoge los diferentes roles que intervendrán en el desarrollo del software, así como las tecnologías usadas.

Calidad: La palabra calidad tiene múltiples significados. La calidad de un producto o servicio es la percepción que el cliente tiene del mismo. Es una fijación mental del consumidor que asume conformidad con un producto o servicio determinado, que solo permanece hasta el punto de necesitar nuevas especificaciones. La calidad es un conjunto de propiedades inherentes a un objeto que le confieren capacidad para satisfacer necesidades implícitas o explícitas.

Delphi: Método para obtener el conocimiento de los expertos respecto a un tema concreto que consiste en: preguntar a unos expertos sobre este tema; reunir las aportaciones en un documento y volverlo a pasar a todos los expertos. Es un procedimiento para obtener una previsión de un grupo de personas expertos en algún aspecto del fenómeno a pronosticar. Se sigue un procedimiento formal y sistemático, en el que las diferencias de opiniones no son discutidas abiertamente, evitando así confrontaciones e influencias psicológicas.

Estándar: Los estándares requieren ser establecidos con el fin de contar con una referencia que permita identificar oportunamente las variaciones presentadas en el desarrollo de los procesos y aplicar las medidas correctivas necesarias. Es una especificación que regula la realización de ciertos procesos o la fabricación de componentes para garantizar la interoperabilidad.

Framework: en el desarrollo de software es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Conjunto integrado de componentes que colaboran para proporcionar una arquitectura reutilizable para una familia de aplicaciones. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Herramienta: Son los ambientes de apoyo necesario para automatizar las prácticas de Ingeniería de Requisitos.

IEEE: Corresponde a las siglas de (Institute of Electrical and Electronics Engineers) en español Instituto de Ingenieros Electricistas y Electrónicos, una asociación técnico-profesional mundial dedicada a la estandarización.

LRP: Corresponde a las siglas de Lista de Reserva del Producto, Es una lista priorizada que define el trabajo que se va a realizar en el proyecto. Cuando un proyecto comienza es muy difícil tener claro todos los requisitos sobre el producto. Sin embargo, suelen surgir los más importantes que casi siempre son más que suficientes para un Sprint (Iteración). Esta lista puede crecer y modificarse a medida que se obtiene más conocimiento acerca del producto y del cliente. Con la restricción de que solo puede cambiarse entre Sprint. El objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible y para esto la lista debe acompañar los cambios en el entorno y el producto.

Modelo: Arquetipo digno de ser imitado que se toma como pauta a seguir.

Módulo: Parte de un programa de ordenador. De las varias tareas que debe realizar un programa para cumplir con su función u objetivo, un módulo realiza una o varias de dichas tareas.

PDSW: Corresponde a las siglas de Proceso de Desarrollo de Software, define quien está haciendo que, cuando, y como alcanzar un determinado objetivo. Su objetivo es construir un producto de software o mejorar uno existente. Un proceso efectivo proporciona normas para el desarrollo eficiente de software de calidad. En consecuencia, reduce el riesgo y hace el proyecto mas predecible.

Plantilla de Historia de Usuario: En esta plantilla se modela las características específicas sistema, así como la forma en que interactúa el sistema con los clientes y viceversa.

Plantilla de Tarea de Ingeniería: Esta plantilla posee una gran importancia, pues permite definir cada una de las actividades que estarán asociadas a las historias de usuario y que permitirán su implementación. También posibilita conocer que programador está asignado a cada tarea, así como el tiempo que se necesita para su realización, lo que facilita la estimación del tiempo que se llevará cada historia de usuario en implementarse, de acuerdo a su complejidad.

Plantilla de Gestión de cambios: La plantilla de gestión de cambio, es el documento que se genera en la fase de Mantenimiento, en la actividad de soporte. En ella se recogen los cambios que se realicen en esta etapa, que pueden ser de corrección, adaptación o mejora. Además se lleva el control de la persona que realiza el cambio, así como de la que lo revisa.

Plantilla de Caso de Prueba de Aceptación: La plantilla de Caso de prueba de aceptación, se genera de la etapa de pruebas. El objetivo de las pruebas de aceptación es validar que un sistema cumple con el funcionamiento esperado y permitir al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento.

Proceso: Secuencia de actividades que tienen un marcado inicio y fin.

SCRUM: Es una metodología para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.

XP: Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y proporcionando un buen clima de trabajo. Se basa en la retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.

Xerberos: es un sistema en proceso de desarrollo que permite detectar amenazas de seguridad como uso indebido de los recursos de la red, acceso ilegal a estaciones o terminales, daños maliciosos y fuga de información sensible. Integra funcionalidades para auditar acciones, procesos y recursos dentro de la empresa brindando notificaciones de posibles modificaciones y actividad indebida en los mismos. También permite realizar actividades de soporte y mantenimiento en estaciones de trabajo gracias a un sistema de clonación y administración remota de imágenes de sistemas operativos permitiendo gestionar de forma centralizada las particiones, las configuraciones, instalación y desinstalación de software en cada estación. Su interfaz amigable y sencilla permite un fácil acceso a cada funcionalidad del sistema y a los reportes.