

Universidad de las Ciencias Informáticas

Facultad 10



**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS.**

**Título: Propuesta de proceso de aseguramiento de la calidad de la distribución
GNU/Linux Nova y adaptación de la metodología SXP.**

Autores: Darianne Acosta Rodríguez

Eglis Barzaga Llano

Tutores: MsC. Maidely Calderon Montero

Ing. Mónica Albo Castro

Ciudad de la Habana

Mayo 2010



*"Lo fundamental es que seamos capaces de hacer cada día algo
que perfeccione lo que hicimos el día anterior..."*

CHÉ

Declaración de Auditoria

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Autor: Darianne Acosta Rodríguez

Autor: Eglis Barzaga Llano

Tutor(a): Maidely Calderon Montero.

Tutor(a): Mónica Albo Castro

Agradecimientos

A mis padres que han estado cada segundo de mi vida a mi lado apoyándome con su infinito amor y sabios consejos para hacer realidad mi sueño. Por estar al tanto, con su ternura y protección y darle el frente a todos los problema, por inculcarme tantos valores positivos, el amor al trabajo, el sacrificio ante las situaciones de la vida, a ser independiente y llevándome siempre por el camino correcto. Les agradezco su apoyo incondicional, su confianza y dedicación.

A mis familiares en especial a mis abuelos, mis tías y mis primos que me han ayudado y brindado su cariño enternecido durante todo este tiempo y siempre han anhelado que yo sea una profesional, hoy se les cumple este sueño.

A Héctor Rodríguez, Andro Manzanet y Manuel Rivero por apoyarme unos de los momento más difícil que depositaron su confianza en mí y en estos momentos estoy atribuyéndoles ese gesto pues ya me he graduado muchas gracias.

A mi compañera de tesis y amiga que estuvimos unidas ante cualquier problema, a su familia por ayudarme y brindarme su apoyo incondicional.

A mi novio por apoyarme, soportarme, por brindarme su amor y dedicación y darme las fuerzas suficientes para salir adelante en los momentos más difíciles por ser un pedacito de mi corazón.

A todas las amistades que he hecho durante toda mi vida y de alguna manera intervinieron en mi formación como toda una profesional que me han apoyado y ayudado en todo momento, les doy las gracias, siempre los llevaré en mi corazón pues la amistad perdura eternamente.

A mi tutora Mónica por su ayuda incondicional, sacrificio y dedicación pues sin ella no hubiera sido posible obtener estos resultados en el trabajo de diploma.

A la Universidad de las Ciencias Informáticas por forjarme, enseñarme y fortalecerme durante estos 6 años de carrera.

A la revolución y a nuestro comandante por darnos la oportunidad de llegar a ser futuros ingenieros informáticos.

Darianne Acosta Rodríguez.

Agradecimientos

Las primeras personas a quienes les debo el haber logrado llegar hasta este momento tan importante en mi vida profesional es a mis padres los cuales desde el primer momento me brindaron su apoyo, confianza, amor y comprensión.

A mi familia que me brindo siempre su ayuda, en partícular a mi hermana y mi cuñado que siempre estuvieron ahí dándome fuerzas para lograr mi sueño y a mis sobrinos los cuales siendo pequeños niños supieron darme confianza para seguir.

Mi compañera de tesis y mejor amiga que siempre fue optimista ante cualquier dificultad, a sus padres por brindarnos apoyo incondicional no solo en los estudios si no también en lo persona.

A mi novio que siempre estuvo ahí en los buenos y malos momentos y a sus padres por ayudarme y apoyarme.

Los amigos que nunca dudaron de mí, y alentaron para que me esforzara cada vez más.

A mi tutora Mónica por su ayuda incondicional, sacrificio y dedicación pues sin ella no hubiera sido posible obtener estos resultados en el trabajo de diploma

A la revolución por darnos la oportunidad de poder llegar a este nivel educativo.

Eglis Barzaga Llano.

Dedicatoria

Les dedico este gran éxito en especial a mis padres Dolores Rodríguez Rojas y Reynaldo P. Acosta Gutiérrez ya que sin su ayuda nada hubiera sido posible, por ser la razón de mí existir, la luz de mis ojos, en fin por ser los padres más adorables del universo.

Darianne Acosta Rodríguez.

Quisiera compartir la satisfacción del resultado de este trabajo con todas aquellas personas que de una manera u otra confiaron en mí y fueron capaces de transmitirlo. En especial a mi papá que siempre me inculcó la idea de esforzarme mucho cuando se quiere lograr algo en la vida sin temor a caerse.

A mi mamá por ser la madre que todos queremos, por ser la persona maravillosa que es y tratar de que yo lograra por encima de todo, mis sueños.

A mi hermana que desde pequeña me apoyó para que tratara de llegar cada vez más lejos y a sus niños por ser los mejores sobrinos.

No por ser el último es el menos importante, a David que sin su paciencia, dedicación, amor, hubiera sido más difícil la meta, por brindarme su apoyo incondicional, por ser para mí esa persona especial que todos queremos en nuestras vidas. Gracias mi amor.

Eglis Barzaga Llano.

Resumen

En Cuba se ha iniciado hace varios años un proceso de migración hacia el software libre. En este, la Universidad de Ciencias Informáticas (UCI) posee gran protagonismo por los logros alcanzados dentro del tema, entre ellos el desarrollo de la distribución GNU/Linux Nova. Al iniciar el desarrollo de la versión 2.0, se comenzó a utilizar la metodología SXP. Sin embargo se presentaron dificultades durante el proceso de desarrollo que afectaron la calidad del producto final. El objetivo que se trazó la presente investigación fue: definir las actividades generales para establecer el aseguramiento de la calidad de la distribución de GNU/Linux Nova y adaptar la metodología de desarrollo de software SXP a estas.

Durante la investigación se realizó una descripción detallada de la situación actual del enfoque de aseguramiento de la calidad, para determinar el modelo de calidad a seguir. Además fue necesario aplicar un diagnóstico a los líderes de línea para conocer los procesos que se llevan a cabo actualmente en el proyecto.

Para poder describir y definir las nuevas prácticas se realizó un estudio detallado del área de procesos PPQA, de CMMI. A partir de estos se describen un conjunto de actividades generales para establecer el aseguramiento de la calidad de la distribución de GNU/Linux Nova. Tomando como punto de partida las prácticas definidas, se hizo la adaptación de la metodología SXP, para lo cual se hizo un estudio profundo de la misma. Finalmente se definieron las adaptaciones de las fases y artefactos de SXP a las prácticas definidas.

Índice

DECLARACIÓN DE AUDITORIA.....	II
AGRADECIMIENTOS	III
AGRADECIMIENTOS	IV
DEDICATORIA	V
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 CALIDAD DE SOFTWARE	6
1.1.1 <i>Concepto de Calidad y Calidad de Software</i>	6
1.1.2 <i>Factores que determinan la calidad del software</i>	10
1.2 MODELOS DE CALIDAD	10
1.2.1 <i>¿Qué son los modelos de calidad?</i>	10
1.2.2 <i>Ventajas del uso de los modelos de calidad</i>	11
1.2.3 <i>Características de los modelos de calidad</i>	12
1.2.4 <i>Estado actual de los modelos de calidad</i>	16
1.3 ASEGURAMIENTO DE LA CALIDAD	18
1.3.1 <i>Concepto de Aseguramiento de la Calidad</i>	18
1.3.2 <i>Enfoque del Aseguramiento de la Calidad Modelo CMMI</i>	18
1.3.3 <i>Enfoque del Aseguramiento de la Calidad Modelo Metrica3</i>	19
1.3.4 <i>Enfoque del Aseguramiento de la Calidad Modelo ISO 15504</i>	20
1.3.5 <i>Enfoque del Aseguramiento de la Calidad Modelo ISO 9003</i>	21
CAPÍTULO 2: DESCRIPCIÓN Y ANÁLISIS DEL PROCESO	24
2.1 REALIZACIÓN DEL DIAGNÓSTICO DE LA SITUACIÓN ACTUAL DEL PROCESO DE DESARROLLO	24
2.2 DESCRIPCIÓN DE LAS PRÁCTICAS DEL ÁREA DE PROCESO PPQA DE CMMI	27
2.3 DEFINICIÓN DE LAS PRÁCTICAS DE ASEGURAMIENTO DE LA CALIDAD PARA EL DESARROLLO NOVA.	34
.....	47
CAPÍTULO 3: APLICACIÓN A UNA METODOLOGÍA.....	47
3.1 DESCRIPCIÓN DE LA METODOLOGÍA SXP.	47
3.3 ADAPTACIÓN DE LOS ARTEFACTOS DE LA METODOLOGÍA SXP A LAS PRÁCTICAS DE ASEGURAMIENTO DE LA CALIDAD DE NOVA. ...	52
REFERENCIAS BIBLIOGRAFICAS	59
BIBLIOGRAFÍA CONSULTADA	62
GLOSARIO DE TÉRMINOS.....	63
ANEXOS	65
ANEXO1: ENCUESTA	65
ANEXO 2: PLANTILLA DE AUDITORÍA.....	68
ANEXO 3: PLANTILLA DE NO CONFORMIDADES	70
ANEXO 4: PLANTILLA DE EVALUACIONES.....	72

INTRODUCCIÓN

El desarrollo de los sistemas de cómputo, computadoras, han facilitado el desempeño del trabajo y la vida de la sociedad. Esto trajo consigo el surgimiento de los sistemas operativos (SO). El surgimiento de los SO creó gran expectativa debido a que es el software de estos sistemas de cómputo, lo cual implica gran importancia. Andrew S. Tanenbaum plantea que: “Encima del SO está el resto del software de sistema y este es la porción del software que se ejecuta en modo kernel o modo supervisor y está protegido por el hardware contra la intervención del usuario. El SO está ahí para administrar todos los componentes de un sistema complejo”¹. Según William Stallings: “Un SO explota los recursos de hardware de uno o más procesadores para brindar un conjunto de servicios a los usuarios del sistema. El SO administra la memoria secundaria y dispositivos de entrada/salida en beneficio de sus usuarios”². Esta importancia de los SO implica que su perfeccionamiento deba mantenerse activo, debido al desarrollo acelerado que posee el hardware de las computadoras.

Los grandes avances de las tecnologías de la información antes mencionados, traen consigo la aparición de la privatización del software. Esto genera una gran polémica, debido a que atenta contra el desarrollo colaborativo que se llevaba a cabo en los inicios de esta ciencia de la computación. Además algunos países, como Cuba, no pueden pagar las altas cifras que se cobran por utilizar software privativo y es necesario estar acorde con el principio de soberanía tecnológica que desean implementar los países de América Latina.

En los años ´90 se finaliza un SO completamente libre llamado GNU/Linux, compuesto por un núcleo central (Linux) y las herramientas GNU generadas hasta entonces. Con el tiempo, desarrolladores, estudiantes universitarios y empresas de todo el mundo comenzaron a adaptarlo a sus necesidades y a incluir nuevas utilidades, lo que ha permitido que del mismo ya se hayan desarrollado varias versiones [1]. Estas no son más que las conocidas distribuciones GNU/Linux, las cuales han ganado popularidad y están a disposición de la comunidad.

En Cuba se ha iniciado hace varios años un proceso de migración hacia el software libre. Este cuenta ya con varias investigaciones, en las cuales la Universidad de Ciencias Informáticas (UCI) posee gran protagonismo por los logros alcanzados dentro del tema. Actualmente y desde hace 5 años, en la facultad 10 de la UCI existe un proyecto encargado de desarrollar la distribución GNU/Linux Nova basado en la meta distribución Gentoo.

¹ Concepto de Sistemas operativos, por Andrew S. Tanenbaum. (1998).

² Concepto de Sistemas operativos, por Stallings. (2009).

El desarrollo de esta distribución cuenta con varias investigaciones, una gran parte llevadas a cabo como trabajos de diploma, entre ellas podemos encontrar:

- ✓ Propuesta de una Línea de Producción de Software para el proyecto Nova Linux.
- ✓ Herramienta gráfica para efectuar la gestión de paquetes en la distribución de GNU/Linux Nova.
- ✓ Sistema para automatizar la generación de paquetes binarios de la distribución Nova.
- ✓ Personalización de Nova para servicios de telefonía digital.
- ✓ Personalización de Nova para competencias de programación en la Educación Cubana.
- ✓ Guía para la migración a software libre en la empresa DIVEP usando la distribución Nova.
- ✓ Selección, Adaptación e Implantación de Portales CMS libres u Open Source. Portal Nova LNX.

De modo que se puede apreciar que se ha ido avanzando en la práctica del desarrollo, sin embargo la distribución como tal se ha implementado de forma artesanal, aplicando el método de Linus Torvalds: “libere rápido y a menudo”. Este se basa en un equipo de trabajo donde cada integrante tiene un propósito particular y enfoque distinto a los demás, donde se lograría un sistema estable y coherente solo a raíz de una serie de artilugios³.

A raíz del inicio del desarrollo de la versión 2.0 de GNU/Linux Nova, en Marzo de 2009, se comenzó a utilizar la metodología SXP. Sin embargo aún así se presentaron dificultades de coordinación entre los procesos y artefactos de esta, contra los que se aplicaban en la práctica, lo cual afectó la calidad del producto final, GNU/Linux Nova v2.0. Por lo que se ha hecho necesario hacer un estudio de los modelos de calidad, que permita definir un conjunto de actividades para el proceso de aseguramiento de la calidad de la distribución GNU/Linux Nova, que además sea aplicable a la metodología definida para el desarrollo.

Partiendo de la problemática anterior se presenta el siguiente **problema a resolver**: El desarrollo de la distribución GNU/Linux Nova no cuenta con un proceso de aseguramiento de la calidad que se adapte a la metodología SXP permitiendo obtener un producto con la estabilidad y calidad requerida por el cliente. Para orientar la investigación se cuenta con el **objeto de**

³ La Catedral y el Bazar por Eric S Raymond.

estudio: el proceso de aseguramiento de la calidad de las distribuciones GNU/Linux y para delimitar el alcance del mismo se cuenta con el **campo de acción:** el proceso de aseguramiento de la calidad de la distribución GNU/Linux Nova.

Para realizar la investigación se trazó el **objetivo general:** definir las actividades generales para establecer el aseguramiento de la calidad de la distribución de GNU/Linux Nova y adaptar la metodología de desarrollo de software SXP a estas. Contando con los **objetivos específicos:**

- ✓ Definir las actividades generales para el proceso de aseguramiento de la calidad de la distribución GNU/Linux Nova.
- ✓ Ajustar la metodología de desarrollo SXP de acuerdo a las actividades generales del aseguramiento de la calidad de la distribución GNU/Linux Nova.

Con este trabajo de diploma se pretende alcanzar el **posible resultado:**

- ✓ Que el equipo de desarrollo de GNU/Linux Nova cuente con un proceso de aseguramiento de la calidad y que la metodología de desarrollo SXP, aplicada para su implementación, este en concordancia con dicho proceso.

Para agilizar y organizar el presente trabajo se definieron las **tareas de investigación:**

- ✓ Estudiar los conceptos de aseguramiento de la calidad tomando como referencia los distintos modelos de calidad de software.
- ✓ Realizar un estudio del arte del aseguramiento de la calidad a los desarrollos de software en el marco internacional y nacional.
- ✓ Identificar las actividades generales para el aseguramiento de la calidad del desarrollo de la distribución GNU/Linux Nova.
- ✓ Adaptar la metodología de desarrollo SXP a las actividades generales definidas para el proceso de aseguramiento de la calidad de la distribución GNU/Linux Nova.

Los **métodos científicos** utilizados en la investigación:

- ✓ Analítico-Sintético: Para la confección de la investigación se realizó el estudio de conceptos que fueron analizados de manera independiente, estudiando minuciosamente cada uno de ellos y comparando el criterio de disímiles autores y las semejanzas entre ellos.

- ✓ Histórico-Lógico: Se realizó un estudio exhaustivo del surgimiento de los procedimientos para el proceso de aseguramiento de la calidad con el afán de llegar a su esencia. Así como una pesquisa de las deficiencias que podría presentar dicho proceso.

La **técnica de recopilación de información** que se utilizó en este trabajo es:

- ✓ Encuesta: Se llevó a cabo una encuesta, para que a través de los integrantes de equipo de trabajo se obtenga información sobre los procesos del desarrollo que se realizan para la implementación del SO.

Capítulo #1: “FUNDAMENTACIÓN TEÓRICA”: En este capítulo se abordan los conceptos esenciales del tema afines con los procedimientos en el proceso de aseguramiento de la calidad. Contiene un estado del arte sobre el aseguramiento de la calidad en el marco nacional, internacional y a nivel de la Universidad de las Ciencias Informáticas.

Capítulo #2: “DESCRIPCIÓN Y ANÁLISIS DEL PROCESO”: En este capítulo se realiza un diagnóstico de cómo se lleva a cabo el desarrollo de la distribución para definir las actividades generales del proceso de aseguramiento de la calidad que deben realizarse.

Capítulo #3: “APLICACIÓN A UNA METODOLOGÍA”: En este capítulo se realiza un análisis de la metodología SXP, para ajustarla a las prácticas para el proceso de aseguramiento de la calidad definidas por la investigación.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Al transcurrir los años, los grandes avances de la informática y la computación se basan directamente en el uso de las computadoras, para de una forma u otra automatizar los procesos en general. Los sistemas operativos surgen con el objetivo de facilitar el trabajo de los usuarios con las computadoras. Estos son el programa (o software) más significativo de un ordenador, permite la operabilidad de estos por los usuarios. En la actualidad encontramos entre las familias más difundidas de SO: la de Windows, la de Macintosh y la de Unix; entre otros.

Dentro de la familia de Unix se destacan como sistemas distintivos del movimiento de software libre, las distribuciones GNU/Linux, siendo Linux el núcleo en el que se basan las mismas, que incluyen además determinados paquetes de software para satisfacer las necesidades de usuarios específicos [2].

Como software al fin un SO y las distribuciones GNU/Linux, cuenta con un proceso de desarrollo que incluye el aseguramiento de la calidad a través de un modelo, este permite que se revisen y corrijan los posibles errores. Al mismo tiempo, posibilita brindarle al usuario un producto con calidad que es lo más importante de dicho proceso. El sentido de la palabra calidad es satisfacer de una manera u otra las necesidades y expectativas de los usuarios. Se plantea que un producto y/o servicio tiene calidad cuando a través de ellos se satisfacen las necesidades del cliente, es decir, es lo que esperaba o más de lo que esperaba el cliente [3].

En el presente capítulo se exponen los diferentes conceptos que, relacionados con el proceso de aseguramiento de la calidad, serán utilizados en toda la investigación. Entre ellos se destacan: procesos, desarrollo de software, metodología de desarrollo, calidad, modelos de calidad y aseguramiento de la calidad, los cuales ayudan a un mejor entendimiento del **problema a resolver**. Como parte de su contenido, se realizará una comparación entre los modelos de calidad a través de los parámetros que permitan adoptar el de mayor eficiencia para desarrollar futuras versiones de la distribución GNU/Linux Nova.

1.1 Calidad de Software

1.1.1 Concepto de Calidad y Calidad de Software

La calidad no es más que el conjunto de propiedades y características de un producto, proceso o servicio que le confieren su amplitud para satisfacer las necesidades establecidas o implícitas. Esta definición ha evolucionado en los últimos años hasta considerar, de acuerdo con G. Taguchi, la calidad como "las pérdidas que un producto o servicio infringe a la sociedad desde su producción hasta su consumo o uso. A menores pérdidas sociales, mayor calidad del producto o servicio". Este enfoque posee la ventaja de incluir no solo los problemas de calidad clásicos (pérdidas sociales debidas a la variabilidad) sino los actuales (pérdidas sociales debidas a los efectos secundarios nocivos, problemas del Medio Ambiente, etc.) [4].

Tomando como referencia uno de los conceptos de calidad emitidos por el Dr. Walter Shewhart, que concebía la misma como "un problema de variación, el cual puede ser controlado y prevenido mediante la eliminación a tiempo de las causas que lo provocan", es una cualidad y propiedad inherente de las cosas, que permite que estas sean comparadas con otras de su misma especie. La definición de calidad nunca puede ser precisa, ya que se trata de una apreciación subjetiva [5].

Existen muchos conceptos y percepciones de lo que es calidad, sin embargo, para este trabajo se toma un resumen de los conceptos plasmados con el cual se trabajará. Definiéndose así la calidad como: todo proceso que sufre un producto para cumplir con las necesidades y expectativas del cliente. Uno de los factores que más influyen en la calidad de los productos de software es la metodología de desarrollo de software aplicada.

Metodología de desarrollo de software

Una metodología de desarrollo de software no es más que un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información [6].

Es además, una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software. La finalidad de una metodología de desarrollo es garantizar la eficacia (p.ej. cumplir los requisitos iniciales) y la eficiencia (p.ej. minimizar las pérdidas de tiempo) en el proceso de desarrollo de software [7].

Una metodología de desarrollo de software no es más que una de las ramas de la ingeniería de software, la cual se encarga de elaborar estrategias de desarrollo de software que promuevan prácticas adaptativas en vez de predictivas; centradas en las personas o los equipos,

orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa del cliente [8].

Además se define como un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software. Donde las tareas no son más que actividades fundamentales en que se dividen los procesos, los procedimientos se refieren a la definición de la forma de ejecutar la tarea, la técnica es el elemento utilizado para aplicar un procedimiento y la herramienta es el material para realizar una técnica, podemos apoyarnos en las herramientas software que automatizan su aplicación.

Actualmente las metodologías de desarrollo de software han alcanzado gran importancia y popularidad en la industria, por lo que aparecen lo que se conoce como características deseables que no le deben faltar. A continuación se muestran algunas de estas:

- ✓ Existencia de reglas predefinidas.
- ✓ Cobertura total del ciclo de desarrollo.
- ✓ Verificaciones intermedias.
- ✓ Planificación y control.
- ✓ Comunicación efectiva.
- ✓ Utilización sobre un abanico amplio de proyectos.
- ✓ Fácil formación.
- ✓ Herramientas CASE.
- ✓ Actividades que mejoren el proceso de desarrollo.
- ✓ Soporte al mantenimiento.
- ✓ Soporte de la reutilización de software.

Después de haber leído algunos libros y hecho una amplia búsqueda en internet se puede decir que existen muchas metodologías de desarrollo de software y entre ellas tienen características en común y propias a su vez [9].

Apoyándose en el análisis realizado sobre las metodologías de desarrollo de software, se pudo concretar que esta no es más que una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software. La finalidad de una metodología de desarrollo es garantizar la eficacia y la eficiencia en el proceso de desarrollo de software.

Desarrollo de software

El proceso de desarrollo de los sistemas de información es estructurado, planificado y controlado en el marco de trabajo que es la ingeniería de software [10]. En esta última, el término fases de desarrollo expresa cómo ha progresado la implementación de un software y cuánto puede requerir.

Cada versión importante de un producto pasa generalmente a través de una etapa en la que se agregan las nuevas características (etapa alfa), después una donde se eliminan errores (bugs) activamente (etapa beta), y finalmente en donde se han quitado todos los bugs importantes (etapa estable). Estas se pueden anunciar y regular formalmente por los que desarrollan el producto. Sin embargo, en ocasiones se utilizan los términos de manera informal para describir el estado de un producto. Normalmente, muchas compañías usan nombres en clave para las versiones antes del lanzamiento de un producto, aunque el producto y las características reales son raramente secretas [11].

De acuerdo con lo planteado anteriormente, se llega a la definición de que desarrollo de software son todas las fases por las que transcurre un software, desde el Inicio hasta las Pruebas, las cuales incluyen un conjunto de procesos para el desarrollo del producto.

Proceso

Un proceso es un conjunto de actividades o eventos que se realizan o suceden, alternativa o simultáneamente, con un fin determinado. Este término tiene significados diferentes según la rama de la ciencia o la técnica en que se utilice [12].

Un proceso, en términos simplísticos en cierto modo, es una instancia de un programa en ejecución sobre un SO [13]. Este puede ser definido como un conjunto de actividades enlazadas entre sí que, partiendo de una o más entradas (inputs) las transforma generando un resultado (output) [14].

Un proceso es un conjunto de:

- ✓ Actividades o eventos que se realizan con un determinado fin.

- ✓ Instrucciones entregadas a la CPU para cumplir una etapa específica señalada por los comandos de algún programa.
- ✓ Recursos y actividades interrelacionadas que transforman elementos de entrada en elementos de salida.
- ✓ Acciones que realiza la organización, transformando insumos, para crear, producir y entregar sus productos ó servicios, tal que satisfagan las necesidades de sus clientes.
- ✓ Serie sistemática de acciones dirigidas al logro de un objetivo.

Un proceso se constituye de:

- ✓ **Que (s)** o recursos, equipos, herramientas, software y hardware.
- ✓ **Entradas** o que se recibe de dónde y de quién.
- ✓ **Métodos** o procedimientos formas, instrucciones, controles.
- ✓ **Quién** o los recursos: gente, destrezas, experiencia.
- ✓ **Salidas** o que se entrega cómo, cuándo a quién se entrega.
- ✓ **Mediciones** sobre objetivos y metas de calidad, seguridad, resultados y rendimientos [15].

Después de haber analizado todo lo referente al concepto de proceso se llega a la siguiente conclusión: un proceso no es más que un conjunto de actividades que se realizan, y que interrelacionadas obtienen un propósito específico, logrando de esta manera satisfacer las necesidades del cliente.

En este epígrafe quedan plasmados los conceptos necesarios para la investigación, así toda persona que desee leer el documento pueda entender los términos tratados y conocer acerca de lo que aborda trabajo.

1.1.2 Factores que determinan la calidad del software.

Hay una serie de factores que contribuyen de manera positiva en el proceso de desarrollo del software y determinan la satisfacción de las necesidades del cliente, calidad. Estos se pueden clasificar en dos grandes grupos:

Según Pressman están los factores que pueden ser medidos directamente y los factores que solo pueden ser medidos indirectamente.

Por su parte McCall plantea que se centran en tres aspectos importantes de un producto de software, las características operativas, la capacidad de soportar los cambios y adaptabilidad a nuevos entornos.

- ✓ Características operativas: Corrección, ¿Hace lo que quiero?; fiabilidad, ¿Lo hace de forma fiable todo el tiempo?; eficiencia, ¿Se ejecutará en mi hardware lo mejor que pueda?; seguridad (Integridad), ¿Es seguro?; facilidad de uso, ¿Está diseñado para ser usado?
- ✓ Capacidad de soportar los cambios: Facilidad de mantenimiento, ¿Puedo corregirlo?; flexibilidad, ¿Puedo cambiarlo?; facilidad de prueba, ¿Puedo probarlo?
- ✓ Adaptabilidad a nuevos entornos: Portabilidad, ¿Podré usarlo en otra máquina?; reusabilidad, ¿Podré reutilizar alguna parte del software?; interoperabilidad, ¿Podré hacerlo interactuar con otro sistema? [18]

Estos factores que determinan la calidad del software hacen que la misma, sea entendida de una mejor manera y posteriormente sea medida con eficacia. Esta debe ser expresada por medio de otros términos que tengan más sentido para el usuario, o en este caso, para el desarrollador [19].

1.2 Modelos de Calidad

1.2.1 ¿Qué son los modelos de calidad?

Modelo de calidad es un conjunto de buenas prácticas para el proceso de desarrollo de software, enfocado en los procesos de gestión y desarrollo de proyectos. [20]. Es además, una herramienta que encamina a las organizaciones a la mejora continua y la competitividad, ofreciendo especificaciones sobre el tipo de exigencias que deben de implementar para poder brindar productos y servicios de alto nivel [21].

De aquí que se puede resumir como modelo de calidad, un conjunto de buenas prácticas para el proceso de desarrollo de software y, una herramienta que encamina a las organizaciones a la mejora continua y la competitividad para obtener un producto de alto nivel.

1.2.2 Ventajas del uso de los modelos de calidad

El uso de los modelos de calidad aporta grandes ventajas para las organizaciones que desarrollan software, entre estas se pueden encontrar:

- ✓ Asegurar el cumplimiento de los requerimientos iniciales solicitados por el cliente, garantizando la satisfacción de los clientes internos y externos.

- ✓ Reducción de los errores de explotación logrando así una mayor fiabilidad del software.
- ✓ Tener una oportunidad para corregir los procesos que se hayan desajustado con el tiempo.
- ✓ Clasificar a las empresas como de clase mundial, logrando que sean más competitivas.
- ✓ Certificar la competitividad internacional requerida para concurrir a todos los mercados.
- ✓ Cambiar la actitud del personal de la empresa, desarrollando y mejorando el nivel y calidad de vida del personal.
- ✓ Generar una cultura organizacional enfocada a cumplir con los requisitos de los clientes.
- ✓ Mejorar continuamente en la calidad de los procesos utilizados, los servicios y los productos, para garantizar tener permanentemente mejores procesos, productos y servicios.
- ✓ Reducir costos en todos los procesos y aumentar la productividad, efectividad y utilidad de la empresa.
- ✓ Tener productos y servicios con valor agregado.
- ✓ Tener criterios de medición e indicadores congruentes con los utilizados en el mercado donde se desenvuelve la empresa, para poderlos comparar con las mejores prácticas y así conocer las fortalezas y debilidades de la empresa, estableciendo las estrategias necesarias para mejorar [22].

Estos modelos se convierten en una mejor comunicación entre usuarios, dirección y técnicos. Mientras que al cliente le place mucho más saber que el software que va a adquirir tiene la calidad requerida, ofreciéndole una mayor confianza en el resultado del trabajo esperado.

1.2.3 Características de los modelos de calidad

En el mundo existen muchos modelos de calidad, estos son de gran importancia para el progreso de la sociedad, pues son la clave fundamental para satisfacer las necesidades del hombre y distribuir en el mercado productos de alta competitividad. Cada modelo de calidad tiene características que lo definen y lo identifican como son: la revisión, control, gestión, autoevaluación, evaluación, planificar, desarrollar y diseñar, entre otras muy particulares de cada uno. Todos los modelos se relacionan entre sí ya que están conectados con el fin de lograr un objetivo específico, la excelencia y la mejora continua de los productos o servicios. En el momento de seleccionar un modelo de calidad como guía para el desarrollo de un software

surgen preguntas como: ¿Cuál será mejor? o ¿Cuál brindara una mejor calidad al producto?
Entre los que existen en la actualidad se pueden encontrar:

- ✓ Modelo Deming (Japón)
- ✓ EFQM (European Foundation for Quality Model).
- ✓ ISO 9000 (International Organization for Standardization).
- ✓ ISO 9003 (International Organization for Standardization).
- ✓ ISO 9126 (International Organization for Standardization).
- ✓ FUNDIBEQ (Modelo Iberoamericano de Excelencia en la Gestión)
- ✓ SPICE (ISO 15504) Software Process Improvement and Capability dEtermination.
- ✓ CMMI v. 1.2(Capability Maturity Model Integration).
- ✓ Modelo de Calidad para Gobiernos Municipales 2005.
- ✓ SMCT (Modelo de Dirección por Calidad 2006-2008).
- ✓ Modelo Shingo-Excellence in Manufacturing (E.U.A).
- ✓ Modelo Malcom Baldrige (E.U. A).
- ✓ BSC (Balanced Scorecard).
- ✓ Métrica3.

De todos los modelos antes expuestos, cada uno se enfoca en fines distintos como: la gestión de la organización, la autoevaluación, productos en general, mejora continua y desarrollo de software. Este trabajo se enfoca en el proceso de aseguramiento de la calidad de un producto de software como es la distribución GNU/Linux Nova. Para regir este proceso es necesario realizar una comparación de los modelos de calidad para productos de software. A continuación se muestran dichos modelos con algunas de las características a tener en cuenta en la elección:

- ✓ **CMMI v1.2:** Este es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software. Las mejores prácticas de CMMI se publican en los documentos llamados modelos. En la actualidad hay tres áreas de interés cubiertas por los modelos de CMMI:
 - Desarrollo.

- Adquisición.
- Servicios.

Este modelo incrementa la satisfacción de las necesidades de los usuarios internos del sistema (trabajadores), permitiendo la elaboración de productos de calidad, dentro del tiempo y costos previstos. CMMI tiene como objetivo el mejoramiento continuo de la calidad de los procesos y productos de una organización y provee una guía para este mejoramiento continuo estableciendo niveles de madurez. La calidad en el proceso de producción y en los productos de software es una exigencia creciente, dado que cada vez es más amplio el uso del software en procesos que son críticos para las organizaciones.

Los beneficios que ofrece CMMI son los siguientes:

- Las actividades de su organización están explícitamente vinculadas a sus objetivos de negocio.
- Su visibilidad en las actividades de la organización es mayor, para ayudar a asegurar que su producto o servicio cumple las expectativas del cliente.
- Se aprende de las nuevas áreas y de las mejores prácticas.

CMMI está siendo adoptada en todo el mundo, respuesta que ha justificado el compromiso del SEI Center [23].

- ✓ **Modelo Métrica3:** Es una metodología de planificación, desarrollo y mantenimiento de sistemas de información, para sintetizar las actividades del ciclo de vida de los proyectos de software dentro de las Administraciones Públicas [24]. Los objetivos de la métrica3 son: proporcionar una medida de los costes en que se incurre durante la realización de un proyecto y comparar los mismos con los beneficios. [25]

Sus ventajas son:

- Valorar la necesidad y oportunidad de acometer la realización del proyecto.
- Seleccionar la alternativa más beneficiosa para la realización del proyecto.
- Estimar adecuadamente los recursos económicos necesarios en el plazo de realización del proyecto.

Esta metodología está basada en el Modelo de Procesos del Ciclo de Vida de Desarrollo, ISO/IEC 12207 (Information Technology - Software Life Cycle Processes)

así como en la norma ISO/IEC 15504, SPICE (Software Process Improvement and assurance standards Capability dEtermination). Está orientada a procesos, descomponiendo cada uno de los procesos en actividades, y éstas a su vez en tareas. [26].

Modelo ISO 15504: Modelo para la mejora y evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software.

El proyecto SPICE en sus inicios tenía tres objetivos principales:

- Desarrollar un borrador de trabajo de un estándar para la evaluación de procesos de software.
- Llevar a cabo los ensayos en la industria, de la norma emergente.
- Promover la transferencia de tecnología, de la evaluación de procesos de software a la industria del software a nivel mundial [27].

Este modelo establece conjuntos predefinidos de procesos, con objeto de definir un camino de mejora para una organización. En concreto, establece 6 niveles de madurez para clasificar a las organizaciones [28].

Actualmente es un estándar internacional que es aplicable a cualquier organización/empresa que quiera conocer y mejorar la capacidad de sus procesos, independientemente del tipo de organización, del modelo de ciclo de vida adoptado, de la metodología de desarrollo y de la tecnología utilizada. El beneficio principal derivado de esta revisión, ha sido la ampliación del alcance del estándar, que ya no restringe su aplicación a los procesos del ciclo de vida del software (recogidos en el estándar ISO/IEC 12207 publicados como anexo de la norma ISO/IEC 15504), sino que puede ser utilizado como un mecanismo de evaluación de cualquier tipo de procesos. [29].

✓ **Modelo ISO 9000-3:** Proviene de un grupo de 5 estándares de administración y aseguramientos de la calidad genéricos, es decir, aplicables a cualquier producto. Son los estándares utilizados para el desarrollo, suministro y mantenimiento del software. Su ámbito de aplicación: [30].

- Desarrollo de Sistemas de información.
- Proceso de Ciclo de Vida.
- Calidad de Software

Es requerida por todas las compañías desarrolladoras de software para:

- Incursionar en el mercado europeo.
- Cubrir las expectativas de los clientes.
- Obtener beneficios de calidad.
- Como estrategia de mercado.
- Para reducir costos de producción.

Algunos de los beneficios que se obtienen con la certificación de la Norma ISO 9000-3 son:

- Mejor documentación de los sistemas.
- Cambio cultural positivo.
- Incremento en la eficiencia y productividad.
- Mayor percepción de calidad.
- Se amplía la satisfacción del cliente.
- Se reducen las auditorías de calidad.
- Agiliza el tiempo de desarrollo de un sistema.

El análisis, de los modelos de calidad, realizado hasta este momento, permite apreciar la importancia de la utilización de estos modelos en las diversas ramas del desarrollo. Lo cual confirma las razones que llevan al incremento de su aplicación a nivel internacional, nacional y en la UCI.

1.2.4 Estado actual de los modelos de calidad.

A nivel internacional:

El mundo globalizado exige cada vez más la aplicación de estándares internacionales que garanticen la calidad de los productos y los servicios. Por esta razón, es necesario que en los procesos de desarrollo de la industria software se incluyan estándares de calidad que permitan certificar los productos y servicios. En el modelo de calidad que se seleccione deberán incluirse instrumentos de medición que permitan calificar cuantitativamente cada una de las características del software o servicio en cuestión. Es importante mencionar, que dependiendo de los distintos tipos de aplicaciones, las métricas podrán variar, pues aunque la mayoría de los

productos y servicios poseen características comunes, cada uno en particular requiere una evaluación específica. [31].

En un mercado globalizado donde las empresas deben innovar y mejorar continuamente para crecer y ser más competitivas, es necesario tener acceso a certificaciones de calidad internacionales, las cuales les dan un respaldo y permiten mantenerse en este mercado. Las certificaciones de calidad en la industria del software ayudan a las empresas a ser más productivas disminuyendo costos y tiempo en su desarrollo.[32].

A nivel Nacional.

El impacto de la industria del software en la nueva economía cubana, es de particular significación, pues se está fomentando la producción y comercialización de software. Se demuestra la necesidad de utilizar metodologías adecuadas para diseñar sistemas con mayor productividad y mejor calidad, así como transmitir los conocimientos de Ingeniería de Software fundamentalmente a los profesionales de la salud vinculados con la computación [33]. Esto último debido a que fue en esta rama de la economía cubana donde se dieron los primeros pasos. Actualmente se han incrementado el número de empresas cubanas desarrollando y comercializando software, como se ha podido apreciar en las Ferias de Informática Habana. Producto de la necesidad de insertarse en el mercado internacional ya se ha comenzado incursionar no solo en la aplicación de normas y estándares sino en la certificaciones de las mismas.

A nivel de universidad.

La Universidad de las Ciencias Informáticas (UCI) es un centro cuya misión es producir software y servicios informáticos, utilizando un nuevo modelo de formación basado en el estudio-trabajo. Actualmente es la mayor empresa productora de software del país por lo que está llevando a cabo un proceso de mejora basado en el modelo CMMI (Capability Model Integration) y utilizando los servicios del SIE Center (Software Industry Excellence Center) del Tecnológico de Monterrey. Este proceso de mejora está encaminado a que la universidad alcance en el 2010 una certificación internacional de nivel 2 de madurez del modelo CMMI. Suceso que la convertiría en la primera empresa cubana certificada con este modelo y una de las pocas en el área del Caribe.

CMMI es un modelo de referencia para el crecimiento de capacidades y madurez de un proyecto de desarrollo de software, que se enfoca tanto en procesos de Administración como de Ingeniería en Sistemas y Software. Con su instauración se espera alcanzar beneficios como:

- ✓ Calendarios y presupuestos predecibles en los proyectos.
- ✓ Mejora dentro del ciclo de vida del desarrollo de software.
- ✓ Mayor productividad.
- ✓ Mayor calidad de los productos y servicios que ofrece la universidad a sus clientes y por ende la satisfacción de los mismos.
- ✓ Mejorar la moral del personal que labora en el centro.

El servicio 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 esté 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 requerimientos del modelo CMMI, estableciendo planes de mejora con los que la organización oriente sus procesos hacia la consecución de sus metas.

Es evidente la necesidad de aplicar un modelo de calidad al desarrollo de un software para el aseguramiento de su calidad. Por ello es necesario a continuación hacer un análisis más profundo de los modelos antes vistos, viendo como enfocan este proceso de asegurar la calidad.

1.3 Aseguramiento de la Calidad

1.3.1 Concepto de Aseguramiento de la Calidad

El aseguramiento de la calidad, se puede definir como el esfuerzo total para plantear, organizar, dirigir y controlar la calidad en un sistema de producción, con el objetivo de dar al cliente mejores productos [34]. Es un sistema y, como tal, es un conjunto organizado de procedimientos bien definidos y entrelazados armónicamente. [35].

El aseguramiento de la calidad es el conjunto de acciones planificadas y sistemáticas, que son necesarias para proporcionar la confianza adecuada de que un producto o servicio satisface los requisitos de calidad, los cuales deben estar sustentados en la satisfacción de las expectativas de los clientes [36].

Tomando en cuenta los conceptos anteriores se puede decir que el más completo para esta investigación es el tercero, ya que se plantea de manera explícita sus objetivos.

1.3.2 Enfoque del Aseguramiento de la Calidad Modelo CMMI

El Modelo CMMI v1.2 lo considera como un área de proceso, que no es más que un conjunto de mejores prácticas relacionadas entre ellas, para satisfacer un conjunto de objetivos importantes dentro de un área. El área de proceso asociada al aseguramiento de la calidad se llama Aseguramiento de la calidad del producto y el proceso (PPQA por sus siglas en inglés). Esta tiene como objetivo fundamental, proporcionar a todo el equipo de trabajo y dirección con una visión objetiva de los procesos y los productos asociados.

En esta área de proceso están presentes un conjunto de actividades planificadas y constantes, requeridas para asegurar que el software cumplirá con ciertos criterios esperados de calidad. Debe planificarse antes de desarrollar el software, nunca después o en el camino [37].

El área de proceso PPQA involucra:

- ✓ Evaluación objetiva de procesos realizados, productos de trabajo y servicios, frente a las descripciones de trabajo, estándares y procedimientos.
- ✓ Identificación y documentación de no conformidades. Proporcionar “feedback” al “staff” de proyecto sobre los resultados de actividades de garantía y de calidad.
- ✓ Aseguramiento del re-direccionamiento de las no conformidades [38].

1.3.3 Enfoque del Aseguramiento de la Calidad Modelo Métrica3

El modelo Métrica3 contiene actividades propias de la interfaz de Calidad, orientadas a verificar la calidad de los productos. Con el objetivo de:

- ✓ Reducir, eliminar y lo más importante, prevenir las deficiencias de calidad de los productos a obtener.
- ✓ Alcanzar una razonable confianza en que las prestaciones y servicios esperados por el cliente o el usuario cumplan sus expectativas. [39].

En cada fase del proyecto se deberán llevar a cabo determinadas actividades: [40].

- ✓ Identificación de las Propiedades de Calidad del Sistema de Información.
- ✓ Establecimiento del Plan de Aseguramiento de Calidad.
- ✓ Adecuación del Plan de Aseguramiento de Calidad a la Solución.

- ✓ Especificación inicial del Plan de Aseguramiento de Calidad.
- ✓ Especificación detallada del Plan de Aseguramiento de Calidad.
- ✓ Revisión del Análisis de Consistencia.
- ✓ Revisión del Plan de Pruebas.
- ✓ Registro de la Aprobación del Análisis del Sistema de Información.
- ✓ Revisión de la Arquitectura del Sistema de Información.
- ✓ Revisión de la Especificación Técnica del Plan de Pruebas.
- ✓ Revisión de los Requisitos de Implantación.
- ✓ Registro de la Aprobación del Diseño del Sistema de Información.
- ✓ Revisión del Código.
- ✓ Revisión de las Pruebas unitarias, de integración y del sistema.
- ✓ Revisión de los Manuales de Usuario.
- ✓ Revisión de la Formación a Usuarios.
- ✓ Registro de la Aprobación del Sistema de Información.
- ✓ Revisión del Plan de Implantación.
- ✓ Revisión de las Pruebas de Implantación.
- ✓ Revisión de las Pruebas de Aceptación.
- ✓ Revisión de las Pruebas de Mantenimiento.
- ✓ Registro de la Aprobación de la Implantación Sistema de Información.
- ✓ Revisión del Mantenimiento.
- ✓ Revisión del Plan de Pruebas de Regresión.
- ✓ Revisión de la Realización de las Pruebas de Regresión

1.3.4 Enfoque del Aseguramiento de la Calidad Modelo ISO 15504

El modelo ISO 15504 es para la mejora y evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software. Tiene como características fundamental establecer un marco para métodos de evaluación, no es un método o modelo en sí. Comprende: evaluación de procesos, mejora de procesos y determinación de capacidades.

Está alineado con el estándar ISO/IEC 12207 que define los procesos del ciclo de vida del desarrollo, mantenimiento y operación de los sistemas de software. ISO forma parte del panel elaborador del modelo CMMI y el SIE Center que mantiene la compatibilidad y equivalencia de este último con ISO 15504 [41].

Proporciona un marco para la evaluación de los procesos de software. Este puede ser utilizado por las organizaciones que participan en la planificación, gestión, supervisión, control y mejora de la adquisición, suministro, desarrollo, operación, evolución y apoyo del software. Proporciona un enfoque estructurado para la evaluación de los procesos de software para los siguientes fines: [42].

- ✓ Comprender el estado de sus propios procesos de mejora de procesos.
- ✓ Determinar la idoneidad de sus propios procesos para una necesidad particular o clase de los requisitos.
- ✓ Determinar la idoneidad de los procesos de otra organización para un contrato particular o clase de contratos.

1.3.5 Enfoque del Aseguramiento de la Calidad Modelo ISO 9003

El modelo ISO 9003 se aplica cuando la conformidad con los requisitos especificados debe ser asegurada por el proveedor solamente en la inspección y ensayos finales. Es necesario enfatizar que los requisitos del sistema de calidad especificados en esta serie de normas UNIT-ISO (9001, 9002 y 9003) son complementarios (no alternativos) a los requisitos técnicos especificados (producto/ servicio).

Se pretende que esta serie de normas se adopten directamente, pero en ciertas ocasiones puede ser necesario adecuarlas a situaciones contractuales específicas. La norma UNIT-ISO 9000 proporciona una guía tanto en la adaptación como en la selección del modelo apropiado de aseguramiento de la calidad, según se indica en UNIT-ISO 9001, UNIT-ISO 9002 y UNIT-ISO 9003 [43].

Uno de los propósitos de este modelo es proveer una adecuada seguridad de que los productos de software y los procesos en el ciclo de vida del proyecto, están conformes con los requerimientos específicos y se ajustan a los planes establecidos. El aseguramiento de la calidad puede ser interno o externo, dependiendo de si la evidencia de la calidad del producto o del proceso se demuestra a la gerencia del proveedor o del cliente. Este proceso implementa varias actividades como: [44].

- ✓ Implementación del proceso: Esta actividad cuenta con diversas tareas tales como: establecer un proceso de aseguramiento de la calidad ajustado al proyecto; elaborar, documentar, implementar y mantener actualizado un plan de ejecución de las actividades y tareas del proceso; implementar normas, metodologías, procedimientos y herramientas para asegurar la calidad; proporcionar al cliente los registros de las actividades y tareas de aseguramiento de la calidad.
- ✓ Aseguramiento del producto: asegurar que todos los planes requeridos por el contrato estén documentados; así como que todos los productos de software usados y la documentación relacionada con ellos cumplan lo establecido. En la preparación de la entrega de los productos, deberá ser asegurado que ellos satisfacen totalmente sus requerimientos contractuales y que son aceptables para el cliente.
- ✓ Aseguramiento del proceso: proporcionar que los procesos del ciclo de vida del producto cumplen con lo establecido y se ajustan a los planes; asegurar que las prácticas internas del diseño del producto, del ambiente de desarrollo y del ambiente de prueba cumplen lo establecido en el contrato.
- ✓ Aseguramiento del sistema de la calidad: certificar actividades adicionales de la gestión de la calidad de acuerdo con las cláusulas de la norma ISO 9001.

El análisis de los conceptos relacionados con el proceso de aseguramiento de la calidad, realizado anteriormente, permite resumir la importancia de este para alcanzar la calidad como punto vital a tener en cuenta durante todo el proceso de desarrollo del software. Con este proceso se espera obtener la máxima calidad y satisfacción del cliente para la próxima versión de la distribución Nova.

Para definir las actividades generales para establecer el proceso de aseguramiento de la calidad de la distribución de GNU/Linux Nova, se tomará como referencia el modelo de calidad CMMI v1.2. Seleccionado por ser el que más se acerca a las necesidades de la distribución y por encontrarse la universidad en estos momentos, en un proceso de mejora dirigido por el SIE Center encaminado a alcanzar una certificación en el nivel 2 de CMMI, lo cual favorece al éxito de esta investigación.

CAPÍTULO 2: DESCRIPCIÓN Y ANÁLISIS DEL PROCESO

Desde los inicios del desarrollo de software se tuvo en cuenta la necesidad de proporcionar la seguridad y la confianza adecuada de un producto a todos los interesados. En el mundo de la informática esta necesidad se convirtió en un proceso indispensable para el ciclo de vida de un producto, denominado Aseguramiento de la calidad de software. El modelo CMMI v1.2 está conformado por 22 áreas de procesos, entre las que se encuentra, el área de Aseguramiento de calidad del proceso y del producto, la cual está compuesta por un conjunto de buenas prácticas, que se tendrán en cuenta durante la investigación.

Para la exitosa selección de las actividades generales del proceso de aseguramiento de la calidad, es indispensable antes realizar una encuesta o diagnóstico a los líderes de línea y de esta manera hacer un estudio de la situación actual los procesos de aseguramiento de la calidad en Nova.

Además se realiza un estudio a profundidad del área de proceso PPQA presente en el modelo de calidad CMMI. Con el resultado del diagnóstico y el estudio de las prácticas que propone el modelo se definen un conjunto de prácticas para el aseguramiento de la calidad del desarrollo de GNU/Linux Nova.

2.1 Realización del diagnóstico de la situación actual del proceso de desarrollo

Para definir la situación actual del proyecto en cuanto a los procedimientos del área de proceso aseguramiento de la calidad. Fue necesario la construcción y aplicación de una encuesta o diagnóstico, realizado a los líderes de línea del proyecto Nova.

En dicho diagnóstico fueron formuladas 13 preguntas que engloban la información necesaria para el desarrollo de la investigación. Las preguntas están orientadas fundamentalmente a conocer si existen los procesos de desarrollo de la distribución, al menos si están definidos formalmente y si se conocen por los propios líderes. [\[Anexo1\]](#)

Luego de haber aplicado el diagnóstico se presentan las cifras los resultados obtenidos en las preguntas realizadas a los líderes de línea. De un total de 6 encuestados un 66,7% no

concuerdan con que se definen planes objetivos (cronogramas) para dar cumplimiento a los requerimientos y un 33,3% cree que si. ¿Existe algún proceso definido para llevar a cabo la gestión del proyecto teniendo en cuenta los cronogramas y otros planes que afecten al mismo?, el 50% dice que si, un 33,3% no sabe y un 16,7% plantea que no.

¿Existe una definición clara de la estructura de los equipos del proyecto y los roles?, un 50% dice que sí y un 50% dice que no. ¿Existe algún proceso definido para asignar los requisitos, responsabilidades y tareas?, el 66,7% plantea que si y un 33,3% dice que no. ¿Existe un proceso defino para la colaboración entre los equipos del proyecto?, en esta pregunta hay 50% de acuerdo y un 50% que no está de acuerdo. ¿Existe un proceso definido para la captura de requerimientos?, un 83,3 %dice que no y el 16,7% que no sabe.

¿Existe un proceso definido para realizar el diseño?, el 66,7% dice que no y 33,3% plantea que sí. ¿Existe un proceso definido para realizar la construcción?, un 50% dice que no, un 33,3% que si y 16,7% dicen no saber. ¿Existe algún proceso definido para llevar a cabo la identificación de los elementos, componentes y/o productos de la configuración?, un 50% dice que existe y un 50% dice que no.

¿Existe algún sistema para llevar a cabo el proceso de gestión de la configuración y cambios?, un 83,3% dice que sí y un 16,7% dice que no. Si utiliza algún estándar, modelo o metodología para realizar las actividades antes mencionadas, 66,7% dice que no y 33,3% dice que sí. ¿Considera que los procesos antes mencionados son suficientes?, 66,7% está de acuerdo y 33,3% no está de acuerdo.

Se puede percibir de los resultados mostrados, que solo se han definido hasta el momento cronogramas genéricos y los de la línea de servidores. A pesar de que existe un proceso definido para llevar a cabo la gestión del proyecto, asumiendo que los cronogramas y otros planes afecten al mismo, pero no ha sido formalizado.

De manera general para el proyecto están definidos los roles y los equipos de desarrollo, aunque se evidencia en los resultados que no es del conocimiento de todos. Se aplica un proceso general para asignar los requisitos, responsabilidades y tareas pero no se encuentra explícitamente descrito, además no es conocido en su totalidad.

Hasta el momento no existía un proceso para la colaboración entre los equipos del proyecto, actualmente se está trabajando en esto, sin embargo aún no es del todo conocido. Además no está definido formalmente un proceso para la captura de requerimientos, aunque se puede decir que se hace un análisis por competencia, debido a

la ausencia de un cliente específico, el equipo determina las necesidades generales del sistema operativo. Tampoco se cuenta con un proceso para realizar el diseño, pues lo que fundamentalmente se define es un tipo de interfaz de usuario (Gráfica, Texto y Alternativa) y se definen aplicaciones para el usuario.

En el proyecto se lleva a cabo un proceso general para realizar la construcción del SO, pero no se encuentra formalmente definido, por lo que no es conocido por todos los involucrados. Se evidencia que no existe un proceso definido para la identificación de los elementos, componentes y/o productos de la configuración, así como el proceso para la gestión. Sin embargo existe un proceso general para llevar a cabo el proceso de gestión de la configuración y cambios aunque no se encuentra formalmente definido.

En el proyecto hay dos de las líneas, Servidor y Escritorio, que utilizan SXP y el resto no tiene definido ningún modelo o metodología a seguir para el proceso de desarrollo. El 67% de los líderes considera que con estos procesos debe ser suficiente para realizar el desarrollo exitoso de la distribución.

Después de haber mostrado todo el análisis sobre los resultados del diagnóstico efectuado a los líderes de línea, se detectaron varios problemas, como se mencionó anteriormente. Estos pueden ser erradicados aplicando las acciones correctivas propuestas como resultado de la auditoría interna, realizada recientemente⁴ y estrategias para mejorar el aseguramiento de la calidad de software de la distribución GNU/Linux Nova. Algunas de las acciones correctivas son las siguientes:

- ✓ Re – planificación del proyecto en función de los objetivos a cumplir y los requisitos con los que debe cumplir el producto/s a desarrollar.
- ✓ Selección y adaptación en el caso necesario de la metodología de desarrollo de software a utilizar.
- ✓ Asignación de responsabilidades a todos los roles del equipo de desarrollo.
- ✓ Cumplimiento de las políticas establecidas.
- ✓ Control constante de lo real contra lo planificado.
- ✓ Control de la calidad del proceso y el producto.

⁴ Yusleydis Fernández del Monte, Acciones correctivas Auditoría interna en Nova, 2009.

- ✓ Uso de una metodología o modelo de desarrollo de software con la finalidad que guie y organice el proceso a seguir.
- ✓ Mejora de procesos a partir de lo establecido en determinadas áreas establecidas por el modelo CMMI: PPQA, REQM, PP, MA.
- ✓ Uso del expediente de proyecto como mecanismo de formalización de procesos que se llevan a cabo de manera habitual.

Estas acciones correctivas se utilizarán para erradicar las no conformidades generadas durante el proceso de aseguramiento de la calidad del desarrollo de GNU/Linux Nova v2.0. Con el propósito de brindar información objetiva a los miembros del proyecto, acerca de los procesos y productos que en él se realizan, para futuros desarrollos.

2.2 Descripción de las prácticas del área de proceso PPQA de CMMI

El modelo CMMI proporciona la oportunidad de evitar o eliminar barreras, consiste en buenas prácticas que se dirigen al desarrollo y actividades de mantenimiento a los productos y/o servicios. Estas prácticas cubren el ciclo de vida del producto. Este modelo de calidad se define a través de áreas de procesos entre las cuales se encuentra el Aseguramiento de la Calidad del Proceso y del Producto (PPQA).

Esta área de procesos involucra:

- ✓ Una evaluación objetiva de los procesos realizados, así como los productos y/o servicios asociados a estos, frente a las descripciones de trabajo, los estándares y procedimientos.
- ✓ Identificación y documentación de las no conformidades, así como el re-direccionamiento de las no conformidades.
- ✓ Proporcionar regeneración al equipo de proyecto sobre los resultados de las actividades de garantía y de calidad.

El área de procesos PPQA se aplica principalmente a las evaluaciones de las actividades y productos de un proyecto, pero también se aplica a otras actividades como las de formación y entrenamiento.

El modelo sugiere que, para las organizaciones pequeñas el rol de asegurador de la calidad podría ser desempeñado total o parcialmente por el propio equipo de desarrollo, con funciones embebidas en el proceso. Asumir este enfoque implicaría entrenar a todo

aquel que va a realizar actividades de control de la calidad. Además las actividades interpretativas deben estar separadas de los involucrados directamente en el desarrollo o mantenimiento del producto. También debe habilitarse un canal de información independiente, para poder informar gradualmente las no conformidades y usar listas de control, las cuales deben estar disponibles para todos los interesados.

Dentro de esta área de procesos se definen una serie de metas y prácticas, las cuales se clasifican en específicas y genéricas. Una meta específica describe las únicas características que deben estar presentes para satisfacer un área de proceso. Una meta genérica es un modelo de componente requerido, se usa en las apreciaciones para ayudar a determinar si un área de proceso está satisfecha. Además describe las características que deben estar presentes para institucionalizar los procesos que implementa un área de proceso. Una práctica, tanto específica como genérica, es la descripción de una actividad que es considerada importante para lograr la meta asociada. Estas describen las actividades que se espera resulten en el logro de las metas específicas de un área de proceso, apoyándose en una serie de subprácticas destinadas al cumplimiento de dichas prácticas.

Las prácticas obtienen como resultado diversos productos de trabajo, los cuales apoyan el objetivo de las subprácticas. A continuación se explican estos productos:

Productos típicos de trabajo	Descripción
Reportes de tendencias de calidad	Este informe recoge en él, las tendencias de calidad existentes en un proyecto, por lo que se agrupan en un mismo documento. Lo utilizan los probadores y los desarrolladores del producto.
Reportes de no conformidades	El reporte de No Conformidades es el documento en el cual se van a archivar los incumplimientos de los requisitos especificados en el proyecto. Este documento define el procedimiento interno a seguir por los desarrolladores del proyecto para el tratamiento de los errores cometidos.

Estado de acciones correctivas	Este informe, es el documento que contiene las no conformidades encontradas, con la respuesta del equipo de desarrollo y el estado en que se encuentran las acciones correctivas.
Reporte de acciones correctivas	Este es el documento en el cual se van a archivar todas las acciones correctivas, en respuesta a las no conformidades.
Informes de auditorías	Documento que muestra los resultados de las auditorías realizadas.
Tendencias de calidad	Informe que recoge los resultados del proceso de evaluación y, cómo influyen estos en el proceso de desarrollo.
Reportes de aseguramiento de calidad.	El reporte del plan de aseguramiento de la calidad, como su nombre lo indica, es el documento en el cual se va a tener el control estricto de la calidad de un proyecto.
Registros de evaluaciones.	Este es un documento que mantiene el registro de las evaluaciones que se han realizado.

Después de haber explicado los productos, se plantean y explican cada una de las metas y prácticas en las cuales estos están presentes. Entre las Metas y Prácticas específicas en el área de proceso en análisis se puede observar:⁵

- ✓ Evaluar objetivamente los procesos y productos de trabajo: Es la unión de los procesos realizados, los productos asociados y los servicios aplicables a la descripción de procesos, estándares y procedimientos evaluados objetivamente. La

⁵ CMMI® for Development. v1.2.

objetividad en la evaluación del aseguramiento de la calidad de procesos y productos es fundamental para el éxito. Esta se logra mediante la independencia y la utilización de criterios.

Prácticas:

- Se debe evaluar la realización de los procesos designados contra las descripciones, estándares y procedimientos de los procesos aplicados.

Subprácticas:

- Promover un ambiente que aliente a los empleados a participar, identificar y reportar problemas de calidad.
- Establecer y mantener criterios para la evaluación.
- Usar los criterios establecidos para realizar la adherencia de los procesos a sus descripciones, estándares y procedimientos.
- Identificar las no conformidades encontradas durante la evaluación.
- Identificar lecciones aprendidas que puedan mejorar los procedimientos para productos y servicios futuros.

Productos típicos involucrados: Reportes de evaluación, Reportes de no conformidades, Acciones correctivas.

- Evaluar objetivamente los servicios y los productos designados frente a las descripciones de procesos, estándares y procedimientos aplicados.

Subprácticas:

- Seleccionar los productos de trabajo que serán evaluados, basado en criterios de selección definidos.
- Establecer y mantener criterios para la evaluación de productos de trabajo.
- Usar criterios establecidos para la realización de las evaluaciones.
- Evaluar los productos de trabajo antes de ser entregados al cliente.
- Evaluar los productos de trabajo en determinados puntos en el momento de su desarrollo.

- Realizar evaluaciones en ejecución o incrementales de los productos de trabajo y/o servicios contra las descripciones de los procesos, estándares y procedimientos.
- Identificar cada no conformidad encontrada durante la evaluación.
- Identificar lecciones aprendidas que puedan mejorar los procesos para futuros productos y/o servicios.

Productos típicos involucrados: Reportes de evaluación, Reportes de no conformidades, Acciones correctivas.

- ✓ Proveer información objetiva: las no conformidades son seguidas objetivamente y son comunicadas, y su resolución es asegurada.

Prácticas:

- Comunicar objetivamente las no conformidades asegurando su resolución, así como los problemas de calidad. Estos últimos reflejan la falta de adherencia entre los procesos aplicados y la descripción de los procesos.

Subprácticas:

- Resolver cada no conformidad con el miembro apropiado del equipo, cuando sea posible.
- Documentar cada no conformidad cuando estas no puedan ser resueltas dentro del desarrollo del proyecto.
- Escalar cada no conformidad que pueda, o no, ser resuelta dentro del desarrollo del proyecto, informar al responsable para actuar con respecto a esta no conformidad.
- Analizar las no conformidades para identificar tendencias de calidad que puedan resolverse.
- Asegurar que los grupos interesados estén al tanto de los resultados de las evaluaciones y de las tendencias de calidad en el tiempo apropiado.
- Revisar periódicamente las no conformidades y tendencias con el responsable de recibir y actuar para esta no conformidad.
- Responsable de recibir y actuar para esta no conformidad.

- Haga seguimiento a la no conformidad hasta que esta sea resuelta.

Productos típicos involucrados: Reporte de acciones correctivas, informes de auditorías, tendencias de calidad.

- Establecer y mantener registros de las actividades de aseguramiento de calidad.

Subprácticas:

- Registrar las actividades del área de proceso de aseguramiento de calidad del producto y del proceso, con el detalle suficiente para que se conozcan su estado y su resultado.
- Revisar el estado y la historia de las actividades del proceso de aseguramiento de calidad.

Productos típicos involucrados: Reportes de aseguramiento de calidad, registros de evaluaciones, reporte de estado de acciones correctivas, reportes de tendencias de calidad [45].

Entre las Metas y Prácticas genéricas en el área de proceso en análisis se pueden encontrar:

- ✓ Lograr las metas específicas: El proceso apoya y habilita las metas específicas logradas en esta área de proceso, transformando los productos identificados como entrada para producir los de salida.

Prácticas

- Realizar las prácticas específicas: Realizar las prácticas específicas del proceso de aseguramiento de la calidad para desarrollar productos y proporcionar servicios que permitan lograr las metas específicas del área de procesos.

Es válido aclarar que esta primera meta genérica es la primera y contempla la mejora continua, con el objetivo de refinar el proceso e ir avanzando a mayores niveles. Con este propósito existen diversas metas, pero esta es la única a incluir en el nivel 2 de CMMI y no es objetivo avanzar aun más.

- ✓ Institucionalizar un proceso administrado: El proceso se institucionaliza como administrado.

Prácticas

- Establecer una política organizacional: Establecer y mantener una política organizacional para la planeación y realización del proceso de aseguramiento de la calidad.
- Planear el proceso: Establecer y mantener el plan para la realización del proceso de aseguramiento de la calidad.
- Proporcionar los recursos: Mantener los recursos adecuados para la realización del proceso de aseguramiento de la calidad, desarrollo de los productos, y proporcionando los servicios del proceso.
- Asignar la responsabilidad: Asignar responsabilidad y autoridad para la realización del proceso, el desarrollo de los productos, y proporcionar los servicios del proceso de aseguramiento de la calidad.
- Entrenar a las personas: Entrenar a las personas realizando o apoyando el proceso de aseguramiento de la calidad según sea necesario.
- Administrar las configuraciones: Ubicación designada de los productos del proceso de aseguramiento de calidad bajo los niveles apropiados de control.
- Identificar e involucrar los trabajadores relacionados: Identificar e involucrar los trabajadores relacionados con el proceso de aseguramiento de la calidad según lo planeado.
- El monitoreo y control del proceso: El monitoreo y control del proceso de aseguramiento de la calidad, contra el plan para la realización del proceso y tomar las acciones correctivas apropiadas.
- Evaluar objetivamente la adhesión: Evaluar objetivamente la adhesión del proceso de aseguramiento de la calidad frente a la descripción del proceso, las normas, procedimientos, y dirección de las no conformidades.
- Repasar el estado con la alta dirección: Repasar las actividades, el estado y los resultados del proceso de aseguramiento de la calidad y la resolución de los problemas con la alta dirección.

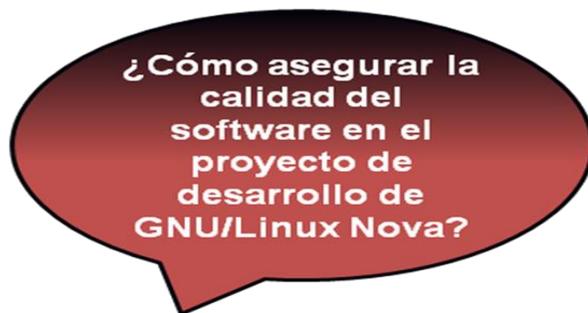
En este epígrafe se describen las metas y prácticas del área de proceso PPQA de CMMI, donde se encuentran detalladas explícitamente todas las condiciones y acciones a cumplir

en esta. Permitiendo definir las prácticas o actividades generales de aseguramiento de la calidad para futuros desarrollos de la distribución GNU/Linux Nova.

2.3 Definición de las prácticas de aseguramiento de la calidad para el desarrollo Nova.

El análisis de las metas y prácticas que propone el modelo CMMI en el área de procesos PPQA desarrollado en el epígrafe anterior, favorece en gran medida la definición de las prácticas del proceso de aseguramiento de la calidad para el desarrollo de la distribución GNU/Linux Nova, objetivo de la presente sección. Las prácticas en PPQA garantizan que los procesos planeados sean implementados.

Para un mejor entendimiento del proceso a describir, es necesario definir los roles generales de un equipo de desarrollo que pudieran estar involucrados. Uno de los roles sería el Líder de proyecto: esta es la persona encargada de disponer y manejar los recursos del proyecto, siempre con el objetivo de mejorar el producto y/o servicio que se brinda. El desarrollador: es el que da forma al producto, es decir, lo crea y/o brinda un servicio. El especialista de calidad: se encarga de probar y verificar la calidad del producto, el servicio y los procesos, en caso de errores informarle al desarrollador o al líder, según corresponda, para que sean rectificados.



A través de las siguientes prácticas o procedimiento, correspondientes a un objetivo o meta:

1. Institucionalizar el proceso de desarrollo de GNU/Linux Nova, esta es una meta genérica que depende de crear habilidades y hábitos de las siguientes prácticas

genéricas. Estas últimas deben ser llevadas a cabo fundamentalmente por el líder del proyecto y un especialista de calidad que lo asesore, a través de:

1.1. Establecer una política organizacional:

- ✓ ¿Qué hacer?: Definir la política organizacional para la realización del proceso de desarrollo de GNU/Linux Nova, esta debe permitir un correcto flujo de trabajo entre los diferentes subproyecto. Pudiera involucrar a todo el equipo con el fin de lograr establecer como parte de esta política la comunicación entre todos los integrantes del proyecto.
- ✓ ¿Cómo?: Definiendo normas de uso de los recursos y comportamiento dentro del proyecto, horarios de trabajo (tiempos de máquinas en el caso de los estudiantes). Además debe definirse cuales, y como, serán los canales de comunicación entre los subproyecto.
- ✓ ¿Cuándo?: Al comenzar el desarrollo de cada versión, como primera actividad de la primera iteración.

1.2. Planear el proceso: Establecer y mantener el plan para la realización del proceso de desarrollo de GNU/Linux Nova.

- ✓ ¿Qué planear?: Los procesos y actividades de todo el ciclo de desarrollo de la distribución.
- ✓ ¿Cómo planear?: Creando un plan general con las fechas y responsables de cada tarea, hito y release candidate de los productos de la distribución. Siempre teniendo en cuenta los riesgos y las afectaciones de los desarrolladores por sus actividades docentes, en lo cual lo debe asesorar el especialista de calidad.
- ✓ ¿Cuándo planear?: El plan debe generarse al inicio del desarrollo de cada versión, no obstante debe ser chequeado y actualizado, ante los cambios y atrasos al finalizar cada iteración del desarrollo.

1.3. Administrar la distribución de los recursos existentes:

- ✓ ¿Qué administrar?: Los recursos materiales, es decir, las computadoras y el personal del proyecto.

- ✓ ¿Cómo administrar?: El líder del proyecto debe mantener una distribución de los recursos adecuada a las necesidades del proceso de desarrollo de cada subproyecto y el avance que este vaya teniendo.
- ✓ ¿Cuándo administrar?: Al comenzar el desarrollo de cada versión, como una de las primeras actividades de la planeación, así mismo puede variar la distribución a medida que se van concluyendo iteraciones.

1.4. Asignar la responsabilidad: Asignar responsabilidades y autoridad a los diferentes roles para la realización del proceso de desarrollo de GNU/Linux Nova y elaboración del producto final.

- ✓ ¿Qué responsabilidad asignar?: El líder del proyecto o de cada subproyecto debe definir los roles y responsabilidades de cada uno de estos.
- ✓ ¿Cómo asignar responsabilidades?: El líder del proyecto o de cada subproyecto asignará un rol a un estudiante, o profesor, en dependencia del conocimiento que posea la persona, así mismo las tareas de desarrollo se asignarán a cada integrante del equipo según el rol que desempeña.
- ✓ ¿Cuándo asignar responsabilidades?: Los roles y responsabilidades de cada uno de estos, debe definirse al inicio de cada versión. Las tareas de desarrollo debe asignarse a cada integrante del equipo al comenzar cada iteración.

1.5. Entrenar a las personas: Entrenar a las personas según sea necesario para realizar y apoyar el proceso de desarrollo de GNU/Linux Nova.

- ✓ ¿Cómo entrenar a las personas?: Conformando e impartiendo un conjunto de cursos, que cada integrante del proyecto debe ir acreditando a medida que va alcanzando habilidades y de acuerdo al rol que desempeña en el proyecto. Esta capacitación debe quedar especificada en un reporte que esté disponible para que todo el proyecto lo conozca.
- ✓ ¿Cuándo asignar responsabilidades?: La capacitación debe realizarse durante la planeación, aunque es posible llevarla conjunto con las primeras iteraciones del desarrollo.
- ✓ ¿Quién asigna responsabilidades?: En el entrenamiento se puede con el tiempo, a medida que se obtenga personal capacitado, involucrar a otros miembros del equipo.

1.6. Identificar e involucrar los trabajadores: Identificar e involucrar los trabajadores relacionados con el proceso de desarrollo de GNU/Linux Nova según lo planeado.

- ✓ ¿Cómo identificar e involucrar los trabajadores?: A través de acciones y de actividades de motivación. Una sugerencia pudiera ser: reconocimiento frente a todo el equipo cuando el desarrollador tiene un logro significativo para el avance del proyecto.
- ✓ ¿Cuándo identificar e involucrar los trabajadores?: Durante todo el ciclo de desarrollo.

1.7. El monitoreo y control del proceso: El monitoreo y control del proceso de desarrollo de GNU/Linux Nova según lo planeado para la realización del mismo y tomar las acciones correctivas apropiadas.

- ✓ ¿Cómo monitorear y controlar el proceso?: A través del cumplimiento correcto de las tareas asignadas, así como del cumplimiento del cronograma definido. El cumplimiento correcto implica que se haya completado con éxito y en el tiempo estimado.
- ✓ ¿Cuándo monitorear y controlar el proceso?: El proceso debe monitorearse y controlarse constantemente, es decir, durante todo el ciclo de vida.

1.8. Repasar el estado con la alta dirección:

- ✓ ¿Qué hacer?: Repasar las actividades, el estado y los resultados del proceso de desarrollo de GNU/Linux Nova, y la resolución de los problemas con la alta dirección.
- ✓ ¿Cómo?: Al iniciar cada iteración se hace una reunión con la alta dirección del proyecto, el líder y los líderes de los subproyectos, en la que se les informa de los resultados de las evaluaciones y se les sugieren acciones correctivas. En esta reunión debe llegarse a consensos sobre las acciones a realizar para mitigar los atrasos, errores y otras afectaciones detectadas durante las evaluaciones.
- ✓ ¿Cuándo?: Al comenzar el desarrollo de cada iteración del proceso de desarrollo.

1.9. Evaluar objetivamente la adhesión: Evaluar objetivamente la adhesión del proceso de desarrollo de GNU/Linux Nova, frente a la descripción del proceso, las normas, los procedimientos, y dirección de las no conformidades. Esta práctica genérica está asociada a dos prácticas específicas que se describen en detalle más adelante.

Lo que sigue a continuación son las prácticas específicas y sus correspondientes subprácticas:

2. Evaluar el proceso de desarrollo de GNU/Linux Nova Evaluar la realización del proceso de desarrollo de GNU/Linux Nova tomando como referencia las descripciones del mismo, así como los estándares y procedimientos aplicados al proceso de desarrollo de GNU/Linux Nova. Este procedimiento debe realizarlo un especialista de calidad o grupo de ellos en dependencia de la complejidad o profundidad de la evaluación.

2.1. Establecer y mantener criterios para la evaluación.

- ✓ ¿Cómo se establecerán los criterios?: A partir de los procesos que se definieron para la aplicación de la metodología, así como los artefactos que permiten documentarlos, se establecen criterios de cumplimiento de dichos procesos y, correcto uso y estado de los artefactos.
- ✓ ¿Cuándo se establecerán los criterios?: Al inicio del desarrollo de la distribución, en este caso al iniciar el desarrollo de GNU/Linux Nova v3.0, y a partir de entonces se usarán estos para futuras versiones, los cambios estarán sujetos a modificaciones en el proceso de desarrollo.
- ✓ ¿Quién establecerá los criterios?: El especialista de la calidad principal en coordinación con el Líder del proyecto.

2.2. Usar los criterios establecidos para evaluar la adherencia de los procesos a sus descripciones, a los estándares y a los procedimientos.

- ✓ ¿Cómo se realizará la evaluación?: Esta evaluación se realizará a través de revisiones y auditorías las cuales se guiarán por una lista de chequeo, donde se indicarán los parámetros a revisar y evaluar en cada una. Estas permitirán encontrar las no conformidades que hasta ese momento se hayan generado en la distribución GNU/Linux Nova.

- ✓ ¿Cuándo se realizará la evaluación?: Al finalizar cada iteración del proceso de desarrollo de la distribución.
- ✓ ¿Quién realizará la evaluación?: Las evaluaciones se llevarán a cabo por el equipo de aseguramiento de la calidad de la distribución GNU/Linux Nova.

2.3. Identificar las no conformidades encontradas durante la evaluación.

- ✓ ¿Cómo se identificarán las no conformidades?: En las revisiones y auditorías a través listas de chequeo, evaluando cada uno de los parámetros definidos en estas. Los parámetros que queden con una evaluación negativa serán los que representarán una no conformidad.
- ✓ ¿Cuándo se identificarán las no conformidades?: Al finalizar cada iteración del proceso de desarrollo de la distribución, durante las revisiones y auditorías.
 - Identificar lecciones aprendidas que puedan mejorar los procedimientos para productos y/o servicios futuros. Estas están asociadas a las no conformidades y a las acciones correctivas que se puedan sugerir para mitigarlas.
- ✓ ¿Cómo se identificarán las lecciones aprendidas?: A partir de las no conformidades identificadas se describirá en un reporte las acciones correctivas sugeridas para mitigarlas.
- ✓ ¿Cuándo se identificarán las lecciones aprendidas?: Al finalizar cada iteración, con el fin de la evaluación y la identificación de las no conformidades. De igual manera en las iteraciones posteriores se actualizará el reporte exponiendo la respuesta a las sugeridas en evaluaciones anteriores y adicionando las nuevas en caso necesario.
- ✓ ¿Quién identificarán las no conformidades?: El propio equipo de aseguramiento de la calidad de la distribución GNU/Linux Nova, que realiza la evaluación.

Debe tenerse en cuenta que las acciones correctivas que se indicarán al final de cada evaluación serán orientadas a solucionar las no conformidades referentes a la calidad, las que se necesiten para dar solución a las no conformidades técnicas deben ser analizadas y definidas en cada equipo de desarrollo.

2.4. Identificar lecciones aprendidas que puedan mejorar los procedimientos para productos y/o servicios futuros. Estas están asociadas a las no conformidades y a las acciones correctivas que se puedan sugerir para mitigarlas.

- ✓ ¿Cómo se identificarán las lecciones aprendidas?: A partir de las no conformidades identificadas se describirá en un reporte las acciones correctivas sugeridas para mitigarlas.
- ✓ ¿Cuándo se identificarán las lecciones aprendidas?: Al finalizar cada iteración, con el fin de la evaluación y la identificación de las no conformidades. De igual manera en las iteraciones posteriores se actualizará el reporte exponiendo la respuesta a las sugeridas en evaluaciones anteriores y adicionando las nuevas en caso necesario.
- ✓ ¿Quién identificarán las no conformidades?: El propio equipo de aseguramiento de la calidad de la distribución GNU/Linux Nova, que realiza la evaluación.

3. Evaluar el producto final: evaluar la calidad del producto final a través de las pruebas y verificando que cumpla con las descripciones del mismo, consiguiendo que el equipo obtenga una clara idea de lo que se quiere y lo que se ha logrado hasta el momento. Esta práctica debe realizarla un especialista de calidad o grupo de ellos en dependencia de la complejidad o profundidad de la evaluación

3.1. Establecer y mantener criterios para la evaluación de productos de trabajo.

- ✓ ¿Qué será evaluado?: Serán evaluados los servicios y/o los productos generados por el proceso de desarrollo de la distribución GNU/Linux Nova. Entre estos se pueden incluir la liberación de las versiones de la distribución en sí, además el servicio principal que brinda el proyecto que consiste en desarrollar personalizaciones ajustadas a las características particulares de una determinada entidad.
- ✓ ¿Cuándo se realizará la evaluación?: La evaluación se realizará siempre que termine un hito en el proceso de desarrollo del producto y/o servicio de la distribución GNU/Linux Nova.
- ✓ ¿Cómo se realizará la evaluación?: Esta se realizará a través de pruebas las cuales mostrarán las no conformidades existentes hasta el momento en el producto y/o servicio de la distribución GNU/Linux Nova.

- ✓ ¿Quién realizará la evaluación?: La evaluación será ejecutada por el equipo de aseguramiento de la calidad de la distribución GNU/Linux Nova.
- 3.2. Utilizar criterios establecidos para la realización de las evaluaciones.
- ✓ ¿Quién establecerá los criterios de evaluación?: El equipo de aseguramiento de la calidad de la distribución GNU/Linux Nova.
 - ✓ ¿Qué criterios?: Los criterios se asociarán a los artefactos de la metodología de desarrollo, que corresponda al proceso de prueba. Además se deben definir criterios de medida para definir la calidad del producto.
- 3.3. Evaluar los productos de trabajo antes de ser entregados al cliente.
- ✓ ¿Cómo evaluar el producto?: El producto se evaluará a través de un proceso de pruebas intensivo.
 - ✓ ¿Cuándo evaluar el producto?: El producto se evaluará durante el proceso de pruebas definido.
 - ✓ ¿Quién evaluará el producto?: Por el equipo de aseguramiento de la calidad.
- 3.4. Evaluar los productos de trabajo en determinados puntos en el momento de su desarrollo.
- ✓ ¿Cómo evaluar el producto?: mediante los diferentes tipos de pruebas definidas en las diferentes etapas del proceso de pruebas.
 - ✓ ¿Cuándo evaluar el producto?: El producto se evaluará en las diferentes fases de desarrollo.
 - ✓ ¿Quién evaluará el producto?: El grupo de aseguramiento de la calidad.
- 3.5. Realizar evaluaciones en ejecución o incrementales de los productos de trabajo y/o servicios contra las descripciones de los procesos, estándares y trabajo y/o servicios contra las descripciones de los procesos, estándares y procedimientos.
- ✓ ¿Qué se evaluará?: La imagen completamente armada y lista para liberación de la distribución GNU/Linux Nova.
 - ✓ ¿Cómo se evaluará?: Las evaluaciones se realizarán observando el funcionamiento del producto basándose en los requerimientos del cliente, estándares definidos para su desarrollo.

- ✓ ¿Cuándo se evaluará?: Las evaluaciones se realizarán durante el funcionamiento del producto en las etapas de pruebas.
- ✓ ¿Quién evaluará?: Fundamentalmente el equipo de especialistas de la calidad, aunque pudieran involucrarse a algunos desarrolladores para obtener mayor cantidad de resultados en menor tiempo.

3.6. Identificar cada no conformidad encontrada durante la evaluación.

- ✓ ¿Cómo se identificarán las no conformidades?: Usando las listas de chequeo, las cuales se definirán de conjunto con los subprocesos del proceso de desarrollo de la distribución.
- ✓ ¿Cuándo se identificarán las no conformidades?: Durante la evaluación, a través de las lista de chequeo.
- ✓ ¿Quién identificará las no conformidades?: El equipo de especialistas de la calidad, aunque pudieran involucrarse a algunos desarrolladores para obtener mayor cantidad de resultados en menor tiempo.

3.7. Identificar lecciones aprendidas que puedan mejorar los procesos para futuros productos.

- ✓ ¿Cómo se identificarán las acciones correctivas?: Usando las listas de chequeo, las cuales se definirán de conjunto con los subprocesos del proceso de desarrollo de la distribución.
- ✓ ¿Cuándo se identificarán las acciones correctivas?: Durante la evaluación, a través de las lista de chequeo.
- ✓ ¿Quién identificará las acciones correctivas?: el equipo de especialistas de la calidad, aunque pudieran involucrarse a algunos desarrolladores para obtener mayor cantidad de resultados en menor tiempo.

4. Proveer información de las no conformidades: documentar la solución de las no conformidades, para tener constancia de los errores erradicados hasta el momento y evitar que se repita alguno de ellos. Acciones realizadas por un especialista de calidad y un desarrollador o grupo de estos.

4.1. Identificar las no conformidades: identificar las no conformidades para que se tengas en cuenta y sean resueltas en el momento adecuado, dando lugar a los encargados de erradicarlas.

- ✓ ¿Qué será identificado?: Serán identificadas las no conformidades generadas por el proceso de desarrollo de la distribución GNU/Linux Nova.
- ✓ ¿Cuándo se identificarán?: Las no conformidades serán encontradas en la terminación de la segunda iteración en la fase de planificación donde se ven las no conformidades documentadas en la iteración anterior durante el proceso de desarrollo de la distribución GNU/Linux Nova.
- ✓ ¿Cómo se identificará?: Las no conformidades se identificarán a través de las evaluaciones del proceso, producto y/o servicio.
- ✓ ¿Quién realizará identificación?: La identificación será ejecutada por el equipo de aseguramiento de la calidad de la distribución GNU/Linux Nova.

4.2. Documentar las no conformidades: documentar las no conformidades, para tener seguridad de su existencia y evitar que ocurra nuevamente en el transcurso del producto y/o servicio, y así darle solución de una manera más rápida.

- ✓ ¿Qué documentar?: Las no conformidades para tener constancia de su existencia, evitar su repetición y poder solucionarlas fácilmente.
- ✓ ¿Cómo documentar las no conformidades?: Estas serán documentadas para tener constancia de su existencia, evitar su repetición y poder solucionarlas fácilmente. De ellas se conocerá su descripción, la descripción del aspecto correspondiente, la etapa en la que se detectó el error y la clasificación de ésta.
- ✓ ¿Cuándo documentar las no conformidades?: Cada vez que se revise se deja el estado anterior y se coloca el nuevo.
- ✓ ¿Quién documenta las no conformidades?: El equipo de desarrollo en correspondencia con el especialista de calidad.

4.3. Documentar las soluciones implementadas: documentar las soluciones implementadas pues si se presenta en reiteradas ocasiones el mismo problema tener documentado como se le dio solución desde la primera vez.

- ✓ ¿Cómo documentar las soluciones?: Actualizando la plantilla de las no conformidades e informándolo al equipo de calidad.

- ✓ ¿Cuándo documentar las soluciones?: Las soluciones serán documentadas luego de recibir respuesta del equipo de desarrollo sobre los resultados de la evaluación, así como en el momento en que se mitigue la no conformidad.
 - ✓ ¿Quién documenta las soluciones?: El equipo de desarrollo en correspondencia con el especialista de calidad. [[Anexo 3](#)]
5. Establecer registros de actividades: establecer y mantener los registros de actividades del proceso de desarrollo de GNU/Linux Nova a través del plan de proyecto, documento en el cual se recogen todas las actividades a desarrollar durante el ciclo de vida del producto. Esta práctica debe ser llevada a cabo por el líder del proyecto y un especialista de calidad.
- 5.1. Registrar las actividades del área de proceso de aseguramiento de calidad del producto y del proceso, con el detalle suficiente para que se conozcan su estado y su resultado.
- ✓ ¿Qué será registrado?: Serán registradas las actividades de área de proceso de aseguramiento de la calidad en la distribución GNU/Linux Nova.
 - ✓ ¿Cuándo se registran?: Serán definidas y documentadas las actividades desde los primeros momentos del proceso de desarrollo de la distribución GNU/Linux Nova.
 - ✓ ¿Cómo se registran?: Estas se registrarán en un documento llamado plan de aseguramiento de la calidad de la distribución GNU/Linux Nova. El mismo recopila una serie de datos que describen como se llevará a cabo el proceso de aseguramiento de la calidad. Este documento incluye en detalle las tareas y responsabilidades de cada integrante del equipo, los estándares y guías utilizadas, las tareas generales de revisiones y auditorías, cronogramas, resolución de problemas y actividades correctivas, proceso de revisión y auditorías, pruebas, herramientas, técnicas y metodologías utilizadas en las actividades del plan, gestión de configuración, registros de calidad y entrenamiento.
 - ✓ ¿Quién registrará las actividades?: Las actividades serán definidas, documentadas y registradas por el líder del proyecto y el especialista de calidad de la distribución GNU/Linux Nova.

5.2. Revisar el estado y la historia de las actividades del proceso de aseguramiento de calidad.

- ✓ ¿Que será revisado?: Se revisarán el estado de cada actividades, si está cumplida o no, y verificar el estado de las mismas.
- ✓ ¿Cómo será revisado?: A través de la verificación del documento.
- ✓ ¿Cuándo será revisado?: Durante el ciclo de vida del producto.
- ✓ ¿Quién revisará?: El especialista de calidad conjuntamente con el líder del proyecto.

Después de haber analizado los resultados arrojados por la encuesta realizada a los líderes de línea de la distribución GNU/Linux Nova, se pudo apreciar la necesidad de definir prácticas para el aseguramiento de la calidad de la misma. Para ello se tuvieron en cuenta las acciones correctivas definidas en una auditoría interna realizada recientemente. También fue necesario indagar sobre las metas y prácticas que propone el modelo CMMI v1.2 para el área de proceso PPQA, las cuales permitieron encaminar la investigación. Logrando obtener las prácticas antes expuestas como parte del proceso de aseguramiento de la calidad para la distribución GNU/Linux Nova.

CAPÍTULO 3: APLICACIÓN A UNA METODOLOGÍA

Unos de los aspectos primordiales dentro de un proyecto de desarrollo de software es el Aseguramiento de la Calidad del Software (SQA). Su importancia se debe a que es un modelo sistemático y planeado de todas las acciones necesarias para proveer la confianza adecuada al cliente, según los requerimientos técnicos establecidos, de cada producto e ítem del proyecto. Los proyectos de software apoyan su elaboración en la aplicación de las metodologías de desarrollo, las cuales como vimos en el capítulo 1 facilitan un conjunto de procedimientos, técnicas, herramientas y un soporte documental.

En este capítulo se retoma el concepto de metodologías de desarrollo de software con el objetivo de hacer un análisis más profundo de la metodología SXP para el desarrollo de GNU/Linux Nova. Luego se describe como se relacionan las prácticas, definidas en el capítulo anterior para el aseguramiento de la calidad en el proyecto GNU/Linux Nova, con la misma.

3.1 Descripción de la metodología SXP.

SXP es una metodología combinada por SCRUM y XP, que brinda una estrategia tecnológica a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva. Fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo.

La metodología SCRUM tiene una forma de gestionar un equipo de manera que trabaje eficazmente y de tener siempre medidos los progresos, de forma que se conozca el estado del proceso y el producto. La metodología XP está encaminada hacia el desarrollo; consiste en una programación rápida o extrema, cuya peculiaridad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Esta utiliza principalmente la técnica de las historias de usuarios para especificar los requisitos del software, las cuales son el único documento de requisitos que se genera en XP, pues detallan lo suficiente para estimar el tiempo que llevará implementarlas⁶.

⁶ Metodología SXP. Wordpad.

SXP está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido cambio de requisitos o requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico, y se orienta a una entrega rápida de resultados y una alta flexibilidad. Esta metodología ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo además seguir de forma clara el avance de las tareas que se realizan, de forma que los jefes pueden ver día a día cómo progresa el trabajo. Las entregas son frecuentes, y existe una refactorización continua, lo que permite mejorar el diseño cada vez que se le añade una nueva funcionalidad.

SXP consta de 4 fases principales:

- ✓ **Planificación-Definición:** Donde se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto. Además se comienza una planificación a partir de la captura y especificación de los requerimientos, pensando desde ya en las entregas rápidas y pequeñas, se realiza un diseño y definición de la arquitectura.
- ✓ **Desarrollo:** Es la fase donde se realiza la implementación y construcción del sistema hasta que esté listo para ser entregado, luego de un conjunto de iteraciones de desarrollo. Se propone el método de la programación en parejas y la integración continua de cada uno de los componentes. Además deben quedar definidos y documentados los estándares de programación a aplicar y los casos de pruebas, tanto las unitarias como las de integración y aceptación.
- ✓ **Entrega:** Fase donde se pone en marcha el producto desarrollado y se genera la documentación necesaria para hacer la entrega al cliente.
- ✓ **Mantenimiento:** En esta se realiza el soporte para los problemas que pueda presentar el software durante su despliegue.

En cada una de estas fases se realizan diversas actividades, tales como: el levantamiento de requisitos, la priorización de la Lista de Reserva del Producto, definición de las Historias de Usuario, diseño, implementación y, pruebas, entre otras; de las cuales se generan artefactos para documentar todo el proceso. Estos se muestran a continuación:

	ARTEFACTOS DEFINIDOS
1	Concepción del sistema

2	Modelo Historias de usuario del negocio
3	Lista de reserva del producto
4	Historias de usuario
5	Lista de riesgos
6	Modelo de diseño
7	Tarea de ingeniería
8	Plan de liberación
9	Estándares de programación
10	Plan de pruebas
11	Caso de prueba de aceptación
12	Manual de usuario
13	Manual de desarrollo
14	Gestión de Cambio

Tabla 1: Artefactos de SXP

Cada uno de estos artefactos cumple una función fundamental en la fase del proceso de desarrollo en que se genera, lo cual se explica seguidamente:

Fase de Planificación – Definición.

En esta fase se tiene como propósito: Establecer la visión, fijar expectativas, y asegurar financiamiento, como entrada: la concepción inicial del producto y se definen las siguientes actividades: Escribir la visión (Plantilla de Concepción del Sistema), presupuesto, Reserva del producto con estimaciones iniciales.

La idea es producir rápidamente versiones del sistema que sean operativas, aunque obviamente no cuenten con toda la funcionalidad pretendida para el sistema, pero sí que constituyan un resultado de valor para el negocio.

- ✓ Concepción del sistema: Es el primer artefacto generado en la fase de Planificación-Definición, este queda elaborado luego de realizarse la entrevista con el cliente.
- ✓ Modelo Historias de usuario del negocio: Es un artefacto que se genera del Juego de la planificación, luego de estar definida la concepción del sistema, se hace mucho más fácil comprender el negocio.
- ✓ Lista de reserva del producto: Es el primer artefacto generado en la etapa de Captura de requisitos, está conformada por una lista priorizada que define el trabajo que se va a realizar en el proyecto.
- ✓ Historias de usuario: Son la técnica utilizada en XP para especificar los requisitos del software, lo que equivaldría a los casos de uso en el proceso unificado.
- ✓ Lista de riesgos: Es el documento que se genera de la actividad de Valoración de riesgos.
- ✓ Modelo de diseño: Es el documento que se genera del Diseño con las metáforas, donde se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.

Fase de Desarrollo.

El propósito de la fase de desarrollo es: Implementar un sistema listo para entrega en una serie de iteraciones de 60 días. (El tiempo puede decrementarse a la medida que se está refinando el producto), la entrada: Lista de Reserva del Producto (LRP) y las actividades: Reunión de planificación de la iteración. Definir la reserva de la iteración. Reuniones de coordinación, revisión de la iteración y la realización de las pruebas de aceptación del sistema.

Las principales ventajas de introducir este estilo de programación son: muchos errores son detectados conforme son introducidos en el código, por consiguiente la tasa de errores del producto final es más baja, los diseños son mejores y el tamaño del código menor, los problemas de programación se resuelven más rápido, se posibilita la transferencia de conocimientos de programación entre los miembros del equipo.

Las pruebas unitarias son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema. Los clientes escriben las pruebas funcionales para cada historia de usuario que deba validarse. El proceso de prueba se está definiendo aún.

- ✓ Plantilla de Tarea de ingeniería: permite definir cada una de las actividades que estarán asociadas a las historias de usuario y que permitirán su implementación.
- ✓ Plan de liberación: Es un documento que se genera de la Junta de planificación.
- ✓ Estándares de programación: Se genera de la etapa de implementación. En él se recoge el estándar utilizado y su explicación.
- ✓ Plan de pruebas: Es lo primero que se realiza cuando el producto va a ser probado.
- ✓ Caso de prueba de aceptación: Se genera en la etapa de pruebas.

Fase de Entrega.

En la fase de entrega el propósito es: Puesta en Operación, las actividades desarrolladas en ella son: Documentación, Entrenamiento, Marketing.

Contiene el cierre de la liberación. Para ingresar a esta fase se debe llegar a un acuerdo respecto a las variables del entorno por ejemplo que los requerimientos fueron completados. El sistema está listo para ser liberado y es en esta etapa en la que se realiza integración, la entrega de las pruebas del sistema y la documentación en general.

Salida: Producto.

Entrega de la documentación: Es una reunión organizada por el Líder del Proyecto, el Cliente y otros interesados donde se hace la entrega de la documentación del sistema implementado.

Entrenamiento: No es más que la actividad realizada para darle capacitación al Cliente y otros interesados que utilizarán el producto.

- ✓ Manual de usuario: Documentos que se genera en la etapa de entrenamiento, sirve de apoyo para la actividad de capacitación.
- ✓ Manual de desarrollo: Es un documento generado en la etapa de entrenamiento.

Fase de Mantenimiento.

✓ Plantilla de gestión de cambio: Es el documento que se genera en la fase de Mantenimiento, en la actividad de soporte.

Se puede apreciar que entre el conjunto de artefactos que propone SXP, no se hace referencia a la utilización de artefactos que propicien la documentación de los procesos investigativos que se realizan a la hora de desarrollar un producto, lo que demuestra que esta metodología no favorece la gestión de las investigaciones [46].

El expediente de SXP aprobado por el centro rector de la calidad en la UCI, Calisoft, plantea que el proceso de desarrollo del producto y/o servicio se realizará a través de ciclos los cuales están integrados por las fases que se definen en la metodología. Un ciclo no es más que lo conocido como una iteración y se plantea que deben durar no más 45 días. Calisoft, en la aprobación de la metodología, propone agregar un conjunto de artefactos entre lo que se encuentran, plan de aseguramiento de la calidad, plan de capacitación y plan de gestión de la configuración. Con la suma de estos artefactos se logra documentar el proceso de desarrollo de cualquier proyecto que utilice el expediente de esta metodología.

3.3 Adaptación de los artefactos de la Metodología SXP a las prácticas de aseguramiento de la calidad de Nova.

Anteriormente se describieron cada una de las fases y los artefactos de la metodología SXP, los cuales tienen una estrecha relación con las prácticas definidas en el capítulo anterior. Como se describió en el epígrafe anterior la mayor parte del proceso de prueba se realiza durante la fase de desarrollo. En el caso particular de Nova se lleva a cabo una parte de este proceso durante la fase de entrega, pues se libera la distribución, en un primer momento, como una versión beta que se sigue probando.

El documento rector del proceso de aseguramiento de la calidad para la distribución GNU/Linux Nova será el plan de aseguramiento de la calidad, dando cumplimiento así a lo establecido por Calisoft.

El proceso de aseguramiento de la calidad definido en el capítulo anterior requiere artefactos que le permitan documentar las políticas organizacionales, la planeación a través de un cronograma, la estructura de los roles y responsabilidades, el proceso de captación, las listas de chequeo, el resultado de las evaluaciones, las no conformidades, las acciones correctivas y las auditorías realizadas.

La metodología SXP entre los artefactos que propone, permite clara y evidentemente realizar esta documentación. Sin embargo se hace necesario añadir o modificar algunos como son:

- ✓ Documentar la política organizacional como parte de la concepción del sistema.
- ✓ Documentar las listas de chequeos a utilizar durante las evaluaciones y/o auditorías.
- ✓ Documentar las evaluaciones que se van realizando con sus resultados, las acciones correctivas referentes a la calidad que surgen y las no conformidades identificadas.

- ✓ Documentar las auditorías realizadas con sus resultados en un informe donde recoja las tendencias de calidad que se van observando y cómo influyen en el proceso de desarrollo. También se debe actualizar las no conformidades de acuerdo con los resultados de la auditoría.

Para lograr esta documentación se sugerirán nuevas plantillas como artefactos del expediente de proyecto a utilizar para el desarrollo de la distribución GNU/Linux Nova. Estas serían las plantillas de auditoría, no conformidades y evaluación, las cuales se podrán ver en [Anexo 2](#), [Anexo 3](#), [Anexo 4](#) respectivamente. Algunas de ellas fueron tomadas del expediente de la metodología RUP definido por Calisoft y otras fueron creadas a partir de las necesidades mencionadas anteriormente.

En cuanto al momento en que debe ejecutarse cada práctica, de acuerdo con las fases que propone SXP, se aprecia que en su mayoría las prácticas están asociadas a las fases de Planificación y Desarrollo, y en menor medida a las de Entrega y Mantenimiento. A continuación se recuerda el propósito de cada una de ellas para relacionar las prácticas del proceso de aseguramiento de la calidad que se debe incluir como parte de las actividades a desarrollar durante esa etapa.

Fase de Planificación y Definición: En esta fase se definen la visión y las expectativas del producto, se asegura el financiamiento y, se realiza la captura y especificación de los requerimientos por parte del cliente y el equipo de desarrollo. De las prácticas definidas para el proceso de aseguramiento de la calidad deben incluirse como actividades de esta fase:

- ✓ Establecer una política organizacional.
- ✓ Planear el proceso. Aunque al iniciar cada iteración o ciclo se debe actualizar los artefactos relacionados con esta actividad.
- ✓ Proporcionar los recursos.
- ✓ Asignar la responsabilidad.
- ✓ Entrenar a las personas. Además de que a medida que vaya siendo necesario adquirir nuevos conocimientos, ya sea en la fase de planificación y definición o en la de desarrollo, pueden incluirse actividades relacionadas con esta práctica.

- ✓ Establecer y mantener criterios para la evaluación. Esta es una de las subprácticas que se corresponde con las prácticas de “Evaluar objetivamente el proceso” y “Evaluar objetivamente el producto final y/o servicio”. Sin embargo los criterios a definir tienen relación con los requerimientos y la definición del proceso, por lo que deben establecerse desde el inicio del proceso de desarrollo.
- ✓ Repasar el estado con la alta dirección. Esta se aplica a partir de la finalización de la primera iteración o ciclo.

Fase de Desarrollo: En esta fase se comienzan a implementar las funcionalidades del producto y se verifica los mismos al finalizar la etapa, concluyendo una iteración, lo cual va a ser definido como un hito.

- ✓ Usar los criterios establecidos para realizar la adherencia de los procesos a sus descripciones, a los estándares y a los procedimientos. Utilizar criterios establecidos para la realización de las evaluaciones.
- ✓ Identificar las no conformidades encontradas durante la evaluación. Esta se lleva a cabo después de la primera iteración o ciclo.
- ✓ Identificar lecciones aprendidas que puedan mejorar los procedimientos para productos y/o servicios futuros.
- ✓ Evaluar los productos de trabajo antes de ser entregados al cliente. El producto será evaluado en la etapa final del proceso de prueba.

Fase de Entrega: Esta fase tiene como propósito la entrega del producto al cliente, en esta etapa se le continúan haciendo pruebas al producto, por la particularidad de que es un sistema operativo.

Fase de Mantenimiento: En esta fase se le da soporte al producto brindándole al cliente una buena calidad para su uso.

Las prácticas que se plantean a continuación se realizan durante todo el proceso de desarrollo.

- ✓ Identificar e involucrar los trabajadores. Se desarrolla durante todo el proceso de desarrollo para de alguna manera mantener motivados a todos los trabajadores.

- ✓ El monitoreo y control del proceso. Es constante y se requiere durante todo el proceso de desarrollo.

A continuación se muestra una tabla que resume la relación entre cada uno de los artefactos de la metodología SXP involucrados con las prácticas definidas para el proceso de aseguramiento de la calidad para la distribución GNU/Linux Nova.

ARTEFACTOS	PRÁCTICAS
<ul style="list-style-type: none"> ✓ Concepción del Sistema 	<ul style="list-style-type: none"> ✓ Establecer una política organizacional. ✓ Proporcionar los recursos. ✓ Asignar la responsabilidad.
<ul style="list-style-type: none"> ✓ Cronograma ✓ Plan de liberación ✓ Lista de Riesgo 	<ul style="list-style-type: none"> ✓ Planear el proceso.
<ul style="list-style-type: none"> ✓ Plan de capacitación 	<ul style="list-style-type: none"> ✓ Entrenar a las personas.
<ul style="list-style-type: none"> ✓ Plan de aseguramiento de la calidad 	<ul style="list-style-type: none"> ✓ El monitoreo y control del proceso. ✓ Establecer registros de actividades.
<ul style="list-style-type: none"> ✓ Plan de evaluaciones ✓ Documento no conformidades 	<ul style="list-style-type: none"> ✓ Evaluar objetivamente el proceso. ✓ Evaluar objetivamente el producto final y/o servicio.
<ul style="list-style-type: none"> ✓ Plan de gestión de cambios 	<ul style="list-style-type: none"> ✓ Proveer información de las no-conformidades.

✓ Plan de auditorias	
✓ Documento no conformidades	

Para llegar a la descripción de la adaptación de los artefactos y las fases de la Metodología SXP a las prácticas de aseguramiento de la calidad de la distribución GNU/Linux Nova, se hizo preciso hacer un estudio de la metodología. Permitiendo dar a conocer los artefactos y fase que interfieren en cada práctica, describiendo como se ajustó la metodología de desarrollo SXP al proceso definido en el capítulo anterior.

CONCLUSIONES

El análisis de los conceptos relacionados con el proceso de aseguramiento de la calidad, realizado, fundamentalmente de los modelos de calidad, permiten resumir la importancia del uso de estos y de la estrategia que se decida para alcanzar el aseguramiento de la calidad, los cuales son puntos vitales a tener en cuenta durante todo el proceso de desarrollo de software. Por lo que se procura tener un elevado interés por todo el equipo de trabajo, y se definen al inicio de cada proyecto de desarrollo de software.

En esta investigación se definen las prácticas a establecer para el proceso de aseguramiento de la calidad de la distribución de GNU/Linux Nova, tomando como referencia el modelo de calidad CMMI v1.2. Seleccionado por ser el que más se acerca a las necesidades de la distribución, además la universidad se encuentra en estos momentos en un proceso de mejora dirigido por el SIE Center encaminado a alcanzar una certificación en el nivel 2 de madurez de CMMI, lo cual favorece al éxito de esta investigación.

Para la definición de las prácticas de aseguramiento de la calidad de la distribución GNU/Linux Nova, se aplicó y analizó los resultados arrojados por una encuesta realizada a los líderes de las líneas de desarrollo de la misma. También se tuvieron en cuenta las acciones correctivas definidas en una auditoría interna realizada recientemente.

Finalmente se obtuvieron 10 prácticas genéricas para dar cumplimiento al objetivo genérico de institucionalizar el proceso y 4 prácticas específicas, todas ellas como parte del proceso de aseguramiento de la calidad de la distribución GNU/Linux Nova. A partir de estas se definen los artefactos necesarios para documentar todo el proceso y estos se ajustan a una metodología de desarrollo.

Durante el análisis de las características de la metodología SXP, se describen las fases que esta define para el proceso de desarrollo de la distribución, así como los artefactos que utiliza para documentar todo esto. Este análisis permitió obtener una adaptación de las prácticas de la metodología al proceso de aseguramiento de la calidad definido, con la adaptación de los artefactos de SXP para lograr una correcta documentación del aseguramiento de la calidad en Nova.

RECOMENDACIONES

- ✓ Este proceso debe ser refinado a partir de su aplicación en la versión 3.0 de GNU/Linux Nova que arrojará experiencia sobre la efectividad del proceso definido.
- ✓ Continuar trabajando en ir incluyendo prácticas que favorezcan el proceso de aseguramiento de la calidad y elevar el nivel hasta lograr llegar al nivel 5 de CMMI.

REFERENCIAS BIBLIOGRAFICAS

- [1].Linux Online Inc., Desarrollo de software, Linux.org, 2010, <http://linux.org/distribuciones>, 25/04/2010.
- [2].Masadelante.com, Familias de sistemas operativos, Masadelante.com, 2010, <http://www.masadelante.com/faqs/sistema-operativo> (23/04/2010).
- [3].Calidad.us, Importancia de la calidad, calidad.us, 2010, <http://www.calidad.us/servicio/consultoria/calidad-y-su-importancia-1/>, 25/04/2010.
- [4].Mgar.net, Calidad, Mgar.net, 2010, <http://mgar.net/soc/isointro.htm> (4/02/010).
- [5].Universidad de las Ciencias Informáticas, Calidad, tesis.uci.cu, 2010, <http://Tesis/Informacion%204-02-010/Definici%C3%B3n%20de%20calidad%20-%20Qu%C3%A9%20es,%20Significado%20y%20Concepto.htm>.25/04/2010.
- [6].Wikipedia.org, Metodología de desarrollo de software, Wiki, 2010, http://es.wikipedia.org/wiki/Metodolog%C3%ADa_de_desarrollo_de_software 25/04/2010.
- [7].Wikipedia.org, Desarrollo de software, wiki, 2010, <http://wikipedia.org/ingeniería>, 25/04/2010.
- [8].Wikipedia.org, Concepto de metodología, wiki, 2010, http://es.wikipedia.org/wiki/Metodolog%C3%ADa_%28ingenier%C3%ADa_de_software%29, 25/04/2010.
- [9]. Scribd, Características deseables de una metodología, Scribd, 2010, <http://www.scribd.com/doc/2050925/metodologias-de-desarrollo-software>, 25/04/2010.[50] ACIS 2010, Metodologías no ágiles, ACIS, 2010, <http://www.acis.org.co/index.php?id=551>, 25/04/2010.
- [10].Wikipedia.org, Desarrollo de software, wiki, 2010, http://es.wikipedia.org/wiki/Fases_del_desarrollo_de_software, 25/04/2010.
- [11]. Marble Station, Desarrollo de software, Marble Station, 2010, <http://www.marblestation.com/?p=644>, 25/04/2010.
- [12].wikipedia.org, Procesos, wiki, 2010, <http://es.wikipedia.org/wiki/Proceso>. (4/02/010)
- [13]. Gestión Empresarial, Procesos, gestionempresarial.info, 2010, http://www.gestionempresarial.info/VerItemProducto.asp?Id_Prod_Serv=28&Id_Sec=8 (4/02/010).

- [14]. Amazon Web Services LLC or its affiliates, Procesos, Amazon Simple Storage Service (Amazon S3), 2010, <http://s3.amazonaws.com/ppt-download/botetn-no-1-julio-de-rbms1627.pdf> (4/02/010).
- [15]. hospedajeydominios.com, Procesos, Hospedaje y Dominios, 2010, http://www.hospedajeydominios.com/mambo/documentacion-manual_linux-pagina-145.html (4/02/010).
- [16]. Definición.de, Procedimientos, Definición.de, 2010, <http://definicion.de/procedimiento/>, 2010.
- [17]. thefreedictionary.com, Procedimientos, thefreedictionary.com, 2010, <http://es.thefreedictionary.com/procedimientos>, 25/04/2010.
- [18]. mitecnológico.com, Factores que determinan la calidad del software, Tecnológico, 2010, <http://www.mitecnologico.com/Main/AnalisisFactoresDeterminanCalidadSoftware>, 25/04/2010.
- [19]. eumed.net, Factores que determinan la calidad del software, eumed.net, 2010, <http://www.eumed.net/libros/2008a/351/Certificacion%20de%20la%20Calidad.htm>, 25/04/2010.
- [20]. Definición.de, Qué es un modelo de calidad, Definición.de, 2010, <http://definicion.de/modelo-de-calidad/>, 25/04/2010.
- [21]. Melvin, Qué es un modelo de calidad, Modelos de la gestión de la calidad, 2010, http://modelosdegestiondelacalidad.blogspot.com/2008/01/que-es-un-modelo-de-calidad_12.html, 12/04/2010.
- [22]. Estructplan Consultora S.A. Argentina, Ventajas del uso de los modelos de calidad, Estructplan on line, 2010, <http://www.estrucplan.com.ar/Producciones/entrega.asp?IDEntrega=815>, 25/04/2010.
- [23]. wikipedia.org, CMMI V.1.2, wiki, 2010, http://es.wikipedia.org/wiki/Capability_Maturity_Model_Integration, 25/04/2010.
- [24]. Carnegie Mellon University, CMMI V.1.2, Software Engineering Intitute, 2010, <http://www.sei.cmu.edu/library/abstracts/reports/06tr008.cfm>, 25/04/2010.
- [25]. Juan Marcos, Modelo de Métrica 3, Modelos de gestión de la calidad de software, 2010, <http://juanmarcosteoria2.blogspot.com/2008/01/modelo-metrica3.html>, 14/04/2010.
- [26]. Melvin, Métrica 3, Modelos de gestión de la calidad, 2010, <http://kybele.escet.urjc.es/documentos/HC/HC4GL2007-T4-MetricaV3-Tecnicas-Compl.pdf>, 25/04/2010.
- [27]. Melvin, Modelos de calidad, Modelos de gestión de la calidad, 2010, <http://modelosdegestiondelacalidad.blogspot.com/25/04/2010>.

- [28].Wikipedia.org, Modelo ISO 15504, wiki, 2010, http://es.wikipedia.org/wiki/Modelo_de_evaluaci%C3%B3n_de_procesos_software_ISO_15504_SPI_CE_de_AENOR, http://es.wikipedia.org/wiki/ISO/IEC_15504, 25/04/2010.
- [29]. Universitat de les Illes Balears, Modelo ISO 15504, Departament de Ciències Matemàtiques e Informàtica, 2010, http://dmi.uib.es/~dmiamp/ESIII/0708_ESIII_SPI_Tema3.pdf, 25/04/2010.
- [30]. Laboratorio Docente de Computación <ldc@ldc.usb.ve>, Modelo ISO 9000-3 LCD: Noticias, 2010, http://www.ldc.usb.ve/~abianc/materias/ci4712/ISO_9000-3.pdf, 25/04/2010.
- [31]. Upiicsa, Actualidad de los Modelos de Calidad a nivel internacional, Upiicsa en línea, 2010, <http://www.revistaupiicsa.20m.com/Emilia/RevEneAbr04/Antonieta1.pdf>, 25/04/2010.
- [32]. Actualidad de los Modelos de Calidad a nivel Internacional. <http://www.utp.edu.co/php/revistas/ScientiaEtTechnica/docsFTP/232815172-176.pdf>.
- [33]. Lilia Ester Rodríguez Chávez, Calidad A nivel nacional, Revista cubana de informática médica, 2010, http://www.rcim.sld.cu/revista_1/articulos_htm/lily.htm, 25/04/2010.
- [33]. CECAM, Actualidad de los Modelos de Calidad a nivel nacional, CECAM, 2010, http://www.rcim.sld.cu/revista_1/articulos_htm/mariarosa.htm, 25/04/2010.
- [34].Wikipedia.org, Aseguramiento de la calidad, wiki, 2010, http://es.wikipedia.org/wiki/Calidad#Aseguramiento_de_la_Calidad, 25/04/2010.
- [35]. Mgar.net, Aseguramiento de la calidad, mgar.net, 2010, <http://mgar.net/soc/isointro.htm>, 15/04/2010.
- [36].Tecnológico, Aseguramiento de la calidad, [www.mitecnologico.com](http://www.mitecnologico.com/Main/SistemaDeAseguramientoDeCalidad), 2010, <http://www.mitecnologico.com/Main/SistemaDeAseguramientoDeCalidad>, 15/04/2010.
- [37]. Bajo Licencia Creative Commons, Enfoque del Aseguramiento de la calidad Modelo CMMI, 2010, http://www.eqsoft.net/presentas/modelos_de_calidad_y_software_libre.pdf, 15/04/2010.
- [38].Melvin, Modelos de gestión de la calidad del software, Enfoque Modelo Métrica 3, Melvin, 2010, <http://modelosdegestiondelacalidad.blogspot.com/2008/01/modelo-de-calidad-no-es-una-metodologia.html>, 15/04/2010.
- [39]. Melvin, Modelos de la gestión de la calidad del software, Enfoque del Aseguramiento de la Calidad Modelo ISO 15504, 2010, <http://modelosdegestiondelacalidad.blogspot.com/2008/01/modelos-gestion-de-calidad-del-software.html>, 15/04/2010.

GLOSARIO DE TÉRMINOS

CPU: Unidad central de procesamiento.

EFQM (European Foundation for Quality Model): Fundación Europea para gestión de la calidad.

ISO 9000 (International Organization for Standardization): Organización internacional de estandarización.

SPICE (ISO 15504) Software Process Improvement and Capability dEtermination: proceso de mejora de software y determinación de capacidades.

CMMI v. 1.2 (Capability Maturity Model Integration): Integración de Modelos de Madurez de Capacidades.

BSC (Balanced Scorecard): Cuadro de mando integral equilibrado.

SIE Center (Software Industry Excellence Center): Centro de industria de software excelente o de excelencia.

(PPQA): Proceso y control de calidad de producto.

Feedback: Realimentación.

Staff: Equipo.

Feedback al staff: Retroalimentación del equipo.

REQM: La gestión de requerimientos.

PP: Presupuesto. Programa de los proyectos de Cooperación de la Unión Europea.

MA: Malcom Baldrige.

Release candidate: Liberación estable.

SQA: Aseguramiento de la Calidad de Proveedores.

MCV: Desarrollo en espiral, utilizado generalmente en la Ingeniería de software. Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades.

SCRUM: Es un "framework" o conjunto de herramientas, 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.

Crystal Methodologies: Metodologías de Cristal.

Dynamic Systems Development Method (DSDM): Método de desarrollo de sistemas dinámicos.

RAD Rapid Application Development: Desarrollo rápido de aplicaciones.

Adaptive Software Development (ASD): Desarrollo de software adaptable.

Feature-Driven Development (FDD): Desarrollo de funciones controladas.

XP: Programación Extrema, es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

Lean Development (LD): Magro desarrollo. Feature-Driven Development (FDD).

Anexo1: ENCUESTA

Estimado usuario:

Somos estudiantes de 5 Año de la Facultad 10 del polo de software libre y estamos realizando la tesis de grado titulada: Propuesta de procedimientos para el proceso de aseguramiento de la calidad de la distribución GNU/Linux Nova. Como parte de la investigación estamos realizando una encuesta y necesitamos que Ud. con sus conocimientos y su experiencia, sobres los procesos llevados a cabo para el desarrollo de dicha distribución nos ayude.

Preguntas.

1) ¿Se definen planes objetivos (cronogramas) para dar cumplimiento a los requerimientos?

Si _____ No _____

2) ¿Existe algún proceso definido para llevar a cabo la gestión del proyecto teniendo en cuenta los cronogramas y otros planes que afecten al mismo?

Si _____ No _____

3) ¿Existe una definición clara de la estructura de los equipos del proyecto y los roles?

Si _____ No _____

4) ¿Existe algún proceso definido para asignar los requisitos, responsabilidades y tareas?

Si _____ No _____

5) ¿Existe un proceso defino para la colaboración entre los equipos del proyecto?

Si _____ No _____

6) ¿Existe un proceso definido para la captura de requerimientos?

Si _____ No _____

a) En caso negativo explicar cómo determinan los requerimientos.

7) ¿Existe un proceso definido para realizar el diseño?

Si _____ No _____

a) En caso negativo diga si al menos realiza algunas de las siguientes actividades:

i) ___ Definir tipo de interfaz de usuario (Grafica, Texto, Alternativa).

ii) ___ Definir aplicaciones para el usuario.

b) ¿Qué otras actividades realizan en la etapa de diseño?

8) ¿Existe un proceso definido para realizar la construcción?

Si _____ No _____

9) ¿Existe algún proceso definido para llevar a cabo la identificación de los elementos, componentes y/o productos de la configuración?

Si _____ No _____

10) ¿Existe algún sistema para llevar a cabo el proceso de gestión de la configuración y cambios?

Si _____ No _____

11) ¿Si usa para realizar las actividades antes mencionadas algún estándar, modelo o metodología?

Si _____ No _____

a) Mencione cual _____

12) ¿Considera que los procesos antes mencionados son suficientes?

Si _____ No _____

a) En caso negativo. ¿Cuáles otros procesos sugiere que faltarían?

13) ¿Que otros procesos se llevan o se pueden llevar a cabo?

Anexo 2: Plantilla de Auditoría

Evaluación de áreas de la organización v2.0

<Nombre del proyecto>

<Nombre del producto>

<Versión>

Control de versiones

Fecha	Versión	Descripción	Autor
<dd/mmm/yy>	<x.x>	<detalles>	<nombre>

Reglas de Confidencialidad

Clasificación: <<Clasificación>>

Este documento contiene información propietaria de la **Universidad de las Ciencias Informáticas y/o <<Empresa Cliente>>**, y es emitido confidencialmente para un propósito específico.

El que recibe el documento asume la custodia y control, comprometiéndose a no reproducir, divulgar, difundir o de cualquier manera hacer de conocimientos público su contenido, excepto para cumplir el propósito para el cual se ha generado.

Estas reglas son aplicables a las **X** páginas de este documento.

1. Introducción
 - 1.1. Propósito
 - 1.2. Alcance
 - 1.3. Definiciones, Acrónimos y Abreviaturas.
 - 1.4. Referencias

[\[Lista de documentos a los que se hace referencia\]](#)

Código	Título
[1]	Documento 1
[2]	Documento 2
[3]	Documento 3

2. Instrumento para la evaluación de áreas a informatizar.

[\[Listas de chequeo, Técnicas de recopilación de información \(entrevistas, revisiones de documentos, observaciones directas, incluyendo lista de elementos a evaluar\)\]](#)

3. Cronograma de la evaluación de áreas a informatizar

Día de la semana	Hora	Elemento a evaluar	Especialistas a participar	Nombres y Apellidos del Especialista.

[Lunes 16-02-2008]	[8:30 a.m.]	[elemento 1]	[Especialista del Departamento de Matemática Básica]	[Ing Juan Pérez Pérez]

4. Informe de resultados

[Descripción de las actividades desarrolladas para la evaluación de la organización, de las sesiones de trabajo, especificar si se cumplió el cronograma, evaluaciones otorgadas a cada elemento evaluado]

5. Fundamentación de los resultados

[Se expone la fundamentación detallada de cada una de las evaluaciones otorgadas a los elementos]

6. Conclusiones de la evaluación

[Pueden ser los resultados porcentuales de los resultados de la evaluación de las áreas a informatizar]

7. Impacto de los resultados obtenidos en la fase de Captura de requisitos

Debilidades	Elemento	Impacto en el software
[debilidad encontrada en la evaluación]	[elemento al que pertenece]	[Descripción del impacto de esta debilidad en el software a desarrollar]

8. Recomendaciones generales

[A continuación se proponen un conjunto de recomendaciones:

1ra variante:

2da variante:

etc.]

9. Recomendaciones técnicas derivadas de la evaluación

[A continuación son presentadas recomendaciones técnicas por cada elemento evaluado:

Elemento 1:

- Recomendación 1
- Recomendación 2

Elemento 2

- ✓ Recomendación 1
- ✓ Recomendación 2]

Anexo 3: Plantilla de no conformidades

DOCUMENTO DE NO CONFORMIDADES V2.0

<Nombre del proyecto>

<Nombre del producto>

<Versión>

Control de versiones

Fecha	Versión	Descripción	Autor
<dd/mmm/yy>	<x.x>	<detalles>	<nombre>

1) Descripción General

[Descripción de Aspectos Generales a tener en cuenta a la hora de realizar el diseño de las pruebas, incidencias en el momento de su desarrollo y otros aspectos relevantes.]

2) Elementos probados

[Descripción general o lista de los Elementos Probados, y otros aspectos importantes a tener en cuenta a la hora analizar las No Conformidades Detectadas.]

3) Elementos no probados y causas

[Descripción de Aspectos Generales a tener en cuenta a la hora de realizar el diseño de las pruebas, incidencias en el momento de su desarrollo y otros aspectos relevantes.]

4) Registro de defectos y dificultades detectados

Elemento	No conformidad	Aspecto correspondiente	Etapas de detección	Clasificación	Estado NC	Respuesta de Equipo Desarrollo
<Nombre del elemento>	<Descripción de la no conformidad>	<Descripción de la no conformidad>	<Etapas de detección>		[Se coloca el estado de la NC]	[Esta columna se comienza a llenar]

Elemento>	de la Norma>	Aspecto correspondiente>	detección del error>	S: Significativa NS: No Significativa R: Recomendación	y la fecha, cada vez que se revise se deja el estado anterior y se coloca el nuevo con la fecha en que se revisó.] RA: Resuelta PD:Pendiente NP:No Procede	a partir de la 2da iteración, y es responsabilidad del equipo de desarrollo, quien especifica la conformidad con lo encontrado o no y en caso de no proceder la no conformidad explica por qué.]

[La NC puede tener solo una de las tres clasificaciones: Significativa, No Significativa o Recomendación]

Anexo 4: Plantilla de evaluaciones

DIAGNÓSTICO V1.0

<Nombre del proyecto>
<Nombre del producto>
<Versión>

Control de versiones

Fecha	Versión	Descripción	Autor
<dd/mmm/yy>	<x.x>	<detalles>	<nombre>

Reglas de Confidencialidad

Clasificación: <<Clasificación>>

Este documento contiene información propietaria de **la Universidad de las Ciencias Informáticas y/o <<Empresa Cliente>>**, y es emitido confidencialmente para un propósito específico.

El que recibe el documento asume la custodia y control, comprometiéndose a no reproducir, divulgar, difundir o de cualquier manera hacer de conocimientos público su contenido, excepto para cumplir el propósito para el cual se ha generado.

Estas reglas son aplicables a las **x** páginas de este documento.

1- Introducción.

- 1.1. Propósito
- 1.2. Alcance
- 1.3. Definiciones, Acrónimos y Abreviaturas.
- 1.4. Referencias

[Lista de documentos a los que se hace referencia]

Código	Título
[1]	Documento 1
[2]	Documento 2
[3]	Documento 3

2- Objetivos del diagnóstico

[Se enuncian los objetivos que se persiguen con la realización del diagnóstico]

3- Responsabilidades de las partes

[Se describen las responsabilidades de ambas partes: la que realiza el diagnóstico y la diagnosticada]

3.1. <Nombre parte 1>

3.2. <Nombre parte 2>

4- Indicadores

[Se enuncian los indicadores para el diagnóstico]

5- Descripción de las entidades a revisar.

5.1. Localización de las entidades

Nro	Nombre	Localización
<i>[Consecutivo]</i>	<i>[Nombre entidad]</i>	<i>[Datos de localización: dirección, estado, provincial, municipio]</i>

5.2. Equipos de diagnóstico por entidades.

5.2.1. Roles y responsabilidades

Rol	Responsabilidades
------------	--------------------------

5.3. Equipos de diagnóstico

Equipo <x>	
Nombre persona	Rol

5.4. Distribución

Nombre de la entidad	Equipo
-----------------------------	---------------

6- Resumen de diagnóstico por entidades

[Resultado del diagnóstico en las entidades atendiendo a los indicadores seleccionados]

6.1. Entidad 1.

6.1.1. Indicador

7- Análisis de los resultados

[Análisis de los resultados obtenidos en el diagnóstico a las entidades. Se realiza un resumen de los aspectos más significativos a tener en cuenta]

8- Conclusiones