

Universidad de las Ciencias Informáticas

Facultad 15



“Propuesta del Plan para la verificación y validación de requisitos en el proyecto ERP- Cuba.”

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas.

Autora: Lisandra Vega Rodríguez

Tutores: Ing. Iliannis Pupo Leyva

Ing. Yeslaine Cortina Blanco

Ing. Dailin Galafet Céspedes

Ciudad de La Habana, Junio del 2010

“Año del 52 Aniversario del Triunfo de la Revolución”



*"Seamos realistas y hagamos lo imposible."
Ernesto "Che" Guevara.*

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Lisandra Vega Rodríguez

Firma del Autor

Iliannis Pupo Leyva.

Firma del Tutor

Yeslaine Cortina Blanco

Firma del Tutor

Dailin Galafet Céspedes

Firma del Tutor

RESUMEN

El aseguramiento de la calidad es una necesidad de toda empresa productora de software con el afán de implantar soluciones propias en medio de la creciente competencia global, y con esto obtener beneficios económicos y sociales. Resulta imprescindible el uso de buenas prácticas durante el desarrollo que conduzcan hacia productos terminados con calidad y con esto lograr la satisfacción del cliente. Se requiere además de procesos que contribuyan a verificar, validar y controlar que dichas prácticas sean realizadas correctamente.

La presente investigación se centra en el estudio de los procesos de Verificación y Validación (V&V) de Requisitos en el proyecto ERP-Cuba como parte del aseguramiento de la calidad del proyecto. El objetivo general de la investigación es elaborar un Plan de (V&V) que facilite la introducción de buenas prácticas en los procesos de Verificación y Validación de Requisitos en el proyecto ERP. Basado en los estudios realizados se decide que la propuesta estará sustentada por la norma IEEE 1012-1998, y se enfocará al Proceso de Desarrollo del ERP- Cuba, incluyendo desde las actividades relacionadas con el estudio preliminar hasta la instalación y aceptación del producto final.

Palabras claves: Aseguramiento de la calidad, Verificación, Validación, Requisitos, ERP, Plan V&V, Proceso de Desarrollo.

TABLA DE CONTENIDOS

RESUMEN	I
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1 Introducción.....	6
1.2 Ciclo de desarrollo del software.....	6
1.3 Calidad del software. Conceptos fundamentales.....	7
1.3.1 Calidad del software.....	7
1.3.2 Buenas prácticas.....	8
1.3.3 El aseguramiento de la calidad del software.....	8
1.3.3.1 Actividades del aseguramiento de la calidad del software.....	9
1.4 Verificación y Validación de requisitos. Definiciones.....	9
1.4.1 Verificación y Validación.....	10
1.4.2 Objetivos del proceso de Verificación y Validación de requisitos.....	11
1.4.3 Ingeniería de requerimientos.....	11
1.4.4 ¿Qué es un requisito o requerimiento?.....	12
1.4.5 Verificación y Validación en el Modelo de Capacidad y Madurez Integrado.....	12
1.5 Métodos de Verificación y Validación de requisitos.....	15
1.5.1 Controles estáticos manuales informales.....	16
1.5.2 Controles estáticos manuales disciplinados.....	16
1.5.2.1 Auditorías.....	17
1.5.2.2 Revisiones técnicas.....	17
1.5.3 Controles dinámicos. Pruebas.....	18
1.5.3.1 Objetivos de las pruebas.....	18
1.5.3.2 Métodos o técnicas que se utilizan para realizar las pruebas.....	18
1.5.3.3 Niveles de pruebas.....	20
1.5.3.4 Herramientas utilizadas en las pruebas.....	20
1.6 Técnicas para la validación de los requerimientos de software.....	21
1.7 Plan de Verificación y Validación de requisitos.....	22

1.9. Conclusiones parciales.....	24
CAPÍTULO 2. PROPUESTA DEL PLAN DE VERIFICACIÓN Y VALIDACIÓN DE REQUISITOS.....	25
2.1 Introducción.....	25
2.2 Descripción del proyecto ERP- Cuba.....	25
2.3 Guía de adopción del estándar.....	26
2.3.1 Nivel de integridad del software	27
2.3.4 Criterios que se emplearán en las tareas.....	28
2.4 El Plan de Verificación y Validación de requisitos para el proyecto ERP (Capital Humano).....	29
2.4.1 Estructura	29
2.4.2 La actividad de Verificación y Validación de requisitos.....	34
2.4.3 La tarea de Verificación y Validación de requisitos	35
2.4.4 Métodos y procedimientos aplicados	37
2.5 Propuesta del Plan de Verificación y Validación de requisitos para el ERP	38
2.6 Conclusiones parciales.....	53
CAPÍTULO 3. VALIDACIÓN DE LA PROPUESTA DEL PLAN DE VERIFICACIÓN Y VALIDACIÓN DE REQUISITOS.....	54
3.1 Introducción.....	54
3.2 El Método Delphi	54
3.3 Aplicación del método	55
3.3.1 Proceso de selección de los expertos	56
3.3.2 Elaboración del cuestionario	57
3.4 Resultados del proceso de aplicación del método	58
3.5 Conclusiones parciales.....	65
CONCLUSIONES	66
RECOMENDACIONES.....	67
REFERENCIAS BIBLIOGRÁFICAS.....	68
GLOSARIO DE TÉRMINOS	71

INTRODUCCIÓN

Los proyectos de desarrollo de software han padecido tradicionalmente problemas de calidad, tanto en el propio proceso de desarrollo como en los productos que entregan. Esta problemática tiene su origen en las habituales desviaciones de plazos y esfuerzo sobre los valores previstos y en la frecuente aparición de fallos durante la implantación y operación de los productos resultantes. Los principales problemas son de naturaleza diferente: por un lado las empresas no dedican recursos ni la atención necesaria para que la actividad tenga éxito y por otro que los esfuerzos que se hacen en su mayoría son reactivos, es decir, cuando ya detectaron un problema en el producto final. Para muchas organizaciones, la información y la tecnología que la soporta representan los activos más valiosos de la empresa; es más, en el competitivo, rápido y cambiante ambiente actual, la gerencia ha incrementado sus expectativas relacionadas con la entrega del servicio y la calidad en cada uno de los sistemas de información. (LEBRÚN et al. 2008).

El aseguramiento de calidad en el software, no es un lema, es una necesidad. Al igual que la Ingeniería de Software, que sugiere la integración de actividades de ingeniería en el desarrollo de software, también es necesario el uso de prácticas que conduzcan hacia productos terminados con calidad. De igual manera, se requiere de procesos que contribuyan a verificar, validar y controlar que dichas prácticas sean realizadas correctamente.

Numerosos estudios relativos al beneficio obtenido a raíz de procesos de mejora, reportan que la implantación de un proceso formal de *Verificación y Validación de Requisitos* supone importantes ventajas para las organizaciones productoras de software (GUZMÁN et al. 2006):

- Se produce un incremento de la satisfacción del cliente al utilizar un software con una cantidad de errores inferior.
- Se incrementa la eficiencia del proceso de desarrollo.
- Se facilita la definición y cumplimiento de los objetivos de calidad.
- Se garantiza que todos los requerimientos presentes en el documento de especificación sigan los estándares de calidad.
- Se verifica que los requerimientos realmente definen el sistema que quiere el cliente, evitando rechazos del producto.

- Se evalúa si la aplicación de software fue desarrollada correctamente y cumple con los requerimientos del usuario.
- El proceso de verificación proporciona pruebas objetivas de que el software, sus artefactos y procesos se ajustan a los requerimientos (integridad, coherencia, precisión, entre otros) en todas las actividades del ciclo de vida (adquisición, suministro, desarrollo, operación y mantenimiento).
- El proceso de validación suministra evidencia que el software, sus artefactos y procesos satisfacen el uso y necesidades del usuario, cumplen los requerimientos del sistema asociados al software al finalizar cada actividad del ciclo de vida y resuelven el problema correctamente.

Cuba trata de convertirse en una de las mayores industrias productoras de software mediante la vinculación de los estudiantes a los proyectos productivos, el capital humano que se forma en la Universidad de las Ciencias Informáticas (UCI) juega un papel fundamental en el cumplimiento de este empeño.

La UCI concibe entre sus principios la automatización de procesos de interés tanto nacionales como internacionales; en este último caso, se apoya en la firma de contratos con diferentes países y entre las expectativas de estos están las relacionadas con la entrega del servicio y la calidad en cada uno de los sistemas de información. Para lograr que la Universidad se adueñe de una posición ventajosa en el competitivo mercado mundial del software, es imprescindible crear productos con calidad, es decir, que satisfagan las necesidades del cliente y que cumpla con las especificaciones requeridas, además que se distingan por ser confiables, mantenibles y flexibles, donde los costos de mantenimiento y perfeccionamiento durante el tiempo de explotación sean cada vez más factibles.

La dirección nacional de nuestro país y con ella, la Universidad de Ciencias Informáticas (UCI), como parte del fortalecimiento de la gestión de las entidades y la informatización de la sociedad crean el proyecto ERP-Cuba; el cual se encuentra enfrascado en obtener un sistema para la Planificación de Recursos Empresariales (Sistema Integral de Gestión-Cedrux) como solución tecnológica a los problemas económicos y sociales, este sistema garantiza la centralización de la información de una empresa.

En el proyecto la ingeniería de requisitos es una de las etapas más críticas dentro de su ciclo de vida, pues en ella se capturan y definen las necesidades que debe cubrir el sistema. La captura de requisitos del usuario, la definición de requisitos y la validación de estos, son algunas de las actividades más

significativas en el desarrollo y la producción del software por el impacto que tiene su resultado en el resto de las fases, ya que de esta etapa se obtienen las necesidades del sistema y el equipo de desarrollo adquiere la información necesaria para desarrollar el producto deseado.

En el desarrollo del proyecto ERP-Cuba se presentan varios problemas, y entre ellos las dificultades que presenta el cliente para establecer explícitamente desde el inicio todos los requerimientos, esto está dado por el gran número de requisitos identificados en un período muy corto, pues el tiempo de retraso de un proyecto es directamente proporcional a la disminución de tiempo y esfuerzo dedicado a las actividades de verificación y validación de requisitos y, más específicamente, a las actividades de prueba de software. Además que el proceso de validación de requisitos se realiza a través de los funcionales, los cuales no son especialistas en los temas propios de los módulos del ERP (Contabilidad, Costos, RRHH, Organización). Esto provoca alta dependencia de analistas, retrasos por cursos de preparación y baja calidad de la validación del sistema.

Además, la ingeniería de requisitos no se desarrolla de una mejor forma, es decir, aún se presentan imprecisiones, lo cual tiene varias causas, la mayoría dadas por la no exploración a fondo de la funcionalidad del sistema. Por tal motivo es que en ocasiones emergen requisitos funcionales en etapas avanzadas de desarrollo, pudiendo haber sido detectados al inicio. De esta forma, sería menos costoso y no provocaría transformaciones en el tiempo previsto para cada actividad; además evitaría retrasos en la entrega del artefacto final. Suele suceder que se comienza a desarrollar un producto tras el levantamiento de los requisitos sin que estos hayan sido aprobados y pudiera ser que cuando esté consumado el producto, el usuario no vea en el mismo, la salida a la totalidad de sus problemas.

La presencia de asuntos como los mencionados, además que la base fundamental de la problemática planteada está en que no existe un documento formal que estandarice y guíe los procesos de Verificación y Validación de Requisitos para el proyecto trae consigo la definición de que el **problema científico** de la investigación sea, ¿Cómo comprobar que los productos cumplan con los requerimientos especificados y que el resultado construido sea el esperado?

El **objeto de estudio** lo constituye el proceso de aseguramiento de la calidad en los proyectos de software de gestión.

El **campo de acción** de la investigación lo constituyen los procesos de Verificación y Validación de Requisitos en el proyecto ERP - Cuba.

Para resolver el problema planteado con anterioridad se propone como **objetivo general**:

Elaborar un Plan de (V&V) que facilite la introducción de buenas prácticas en los procesos de Verificación y Validación de Requisitos en el proyecto ERP- Cuba.

Para el logro de este objetivo general se plantearon los siguientes **objetivos específicos**:

1. Incorporar nuevas actividades de Verificación y Validación de Requisitos a las realizadas en el proceso de desarrollo del ERP-Cuba, según las normas internacionales existentes.
2. Conformar la propuesta de un Plan de V&V que incluya buenas prácticas de Verificación y Validación de Requisitos ajustado a las características del proyecto.
3. Validar la propuesta de solución.

Las **tareas de la investigación** que se llevarán a cabo para dar cumplimiento a los objetivos trazados quedan descritas como sigue:

1. Identificar las actividades de Verificación y Validación de Requisitos que se realizan en el proceso de desarrollo del ERP - Cuba, así como las posibles mejoras a estas.
2. Identificar en normas internacionales nuevas actividades y tareas de Verificación y Validación de Requisitos que se deben incorporar a las existentes en el ERP-Cuba.
3. Analizar métodos V&V de Requisitos que se aplican a lo largo del ciclo de vida de un software y que se puedan ajustar al entorno del proyecto.
4. Confeccionar sobre la base de las buenas prácticas de Verificación y Validación de Requisitos la propuesta del Plan de V&V de Requisitos.
5. Validar el Plan de V&V de Requisitos mediante el Criterio de Expertos.

Se parte de la siguiente **idea a defender**: con la creación de un Plan de V&V de Requisitos para el proyecto ERP-Cuba, disminuye la posibilidad de lograr productos con defectos y se garantizará que estén acorde con los requerimientos del cliente.

La investigación está estructurada en tres capítulos:

En el Capítulo I se tratarán los temas que proporcionan la fundamentación teórica de la investigación donde se enuncian definiciones relativas al ciclo de vida del software, calidad del software y

aseguramiento de la calidad. También se abordan los contenidos relacionados directamente con los procesos de Verificación y Validación.

En el Capítulo II se describe la solución propuesta, donde partiendo de la caracterización general del proyecto ERP, se explican todos los aspectos concernientes a la creación del Plan de Verificación y Validación de Requisitos. Se detallan los elementos relacionados con su estructura, el papel que juegan actividades y tareas incluyendo los métodos y procedimientos que en estas se aplican.

En el Capítulo III se muestra el proceso de validación de la propuesta utilizando el Método Delphi, detallando la manera en que se realizó la selección de los expertos y elaboración del cuestionario aplicado. Se explica además la tabulación de los datos obtenidos de la aplicación de este cuestionario al Panel de Expertos.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se realiza un estudio del estado de los elementos fundamentales que componen un Plan de Verificación y Validación de Requisitos, que servirán como base de conocimiento para entender la propuesta de solución a la problemática que trata esta investigación. Además, enfocará la atención en las herramientas y técnicas fundamentales para la realización de pruebas y auditorías utilizadas en el proyecto ERP-Cuba con el objetivo de garantizar confiabilidad y mantener la integridad de los datos.

1.2 Ciclo de desarrollo del software

Cuando se trabaja para construir un producto o un sistema, es importante seguir una serie de pasos predecibles –un mapa de carreteras que le ayude a obtener el resultado oportuno de calidad-. El mapa de carreteras a seguir es llamado proceso del software. (PRESSMAN, 2002)

Un **proceso** define quien está haciendo qué, cuándo, y cómo alcanzar un determinado objetivo. En la Ingeniería de Software el objetivo es construir un producto software o mejorar uno existente. Durante esta evolución debería limitar su alcance, en un momento del tiempo dado, a las realidades que permitan las tecnologías, herramientas, personas y patrones de la organización. (QUISPE-OTAZU, 2007)

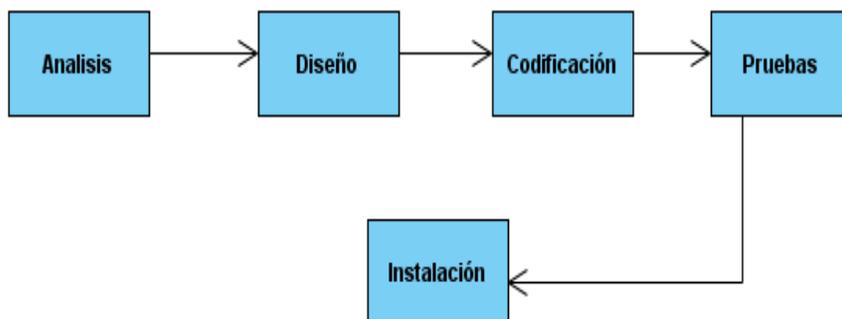


Figura 1: Representación del proceso de desarrollo del software

Algunas definiciones sobre el ciclo de vida de un software plantean lo siguiente:

- ✓ Es el período de tiempo que comienza cuando un producto de software es concebido y termina cuando ya no está disponible para su uso. El ciclo de vida del software incluye las fases de Conceptualización, Requerimientos, Diseño, Implementación, Prueba, Instalación y Chequeo, Operación y Mantenimiento, y en ocasiones la fase de la retirada del software. (IEEE, 1990)
- ✓ El ciclo de vida del software consta de cuatro fases: inicio, elaboración, construcción y transición, con cada fase subdividida en iteraciones. Cada ciclo concluye con una versión de producto para los clientes.(BOOCH et al. 2000)

En el desarrollo de un software el proceso a seguir para su completa concepción es muy importante, ya que para que este tenga la calidad requerida se deben seguir un conjunto de acciones, las cuales forman parte del proceso de desarrollo del producto y lo guían hasta su objetivo final. Entonces se puede decir que un proceso está formado por una serie de pasos, actividades o acciones que dirigen el desarrollo de cualquier actividad hacia un objetivo.

1.3 Calidad del software. Conceptos fundamentales

1.3.1 Calidad del software

Con el transcurso de los años el concepto de calidad se ha venido perfeccionando, ampliando sus objetivos y variando su orientación, se puede decir que ha tomado una importancia creciente al transitar desde un mero control o inspección hasta convertirse en uno de los pilares imprescindibles en la estrategia global de una organización. Ha sufrido una transformación muy importante, pues ha pasado de la simple idea de realizar una verificación hasta tratar de asegurar calidad desde el inicio.

Roger Pressman hace hincapié además en 2 puntos importantes para definir la calidad de software, estos son: (PRESSMAN, 2002)

- Los requisitos del software son la base de las medidas de la calidad. La falta de concordancia con los requisitos es una falta de calidad.
- Los estándares especificados definen un conjunto de criterios de desarrollo que guían la forma en que se aplica la ingeniería del software.

Para la investigación, y los objetivos de este estudio teórico, todas las definiciones referenciadas anteriormente sobre la calidad del software tienen significación pero la más general y completa la constituye la dada por Roger Pressman, con la cual coincide la autora de este trabajo. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad y es esencial para el desarrollo exitoso de un software que esta se asegure y controle.

1.3.2 Buenas prácticas

En primer lugar se entiende por buena práctica toda actuación llevada a cabo que signifique la creación de servicios públicos de calidad o mejore significativamente una situación insatisfactoria. “Buenas prácticas” significa que “existe un acuerdo en que la correcta aplicación de las habilidades, herramientas y técnicas puede aumentar las posibilidades de éxito de una amplia variedad de proyectos”. El equipo de dirección de un proyecto determina lo que es apropiado para el suyo, pues los conocimientos de las “buenas prácticas” no implican que se apliquen de manera uniforme en todos.

A modo de conclusión se considera que las buenas prácticas son un conjunto de acciones que una empresa realiza para mejorar la calidad de lo que construye y pueda superar las expectativas del cliente. En la investigación se ve reflejado en actividades con sus respectivas tareas con las cuales se pretende mejorar el presente y por lo tanto puede ser un modelo o norma para un determinado sistema.

1.3.3 El aseguramiento de la calidad del software

La Garantía de Calidad del Software (SQA, Software Quality Assurance, GCS, Gestión de la Calidad de Software) es una actividad de protección que se aplica a cada paso del proceso del software. Comprende procedimientos para la aplicación efectiva de métodos y herramientas, revisiones técnicas formales, técnicas y estrategias de pruebas, procedimientos de garantía de ajuste a los estándares y mecanismos de medida e información. El aseguramiento de la calidad persigue varios objetivos como son:

- ✓ Revisar los productos y la documentación.
- ✓ Verificar el cumplimiento de las especificaciones de los requisitos.
- ✓ Garantizar que el proceso se lleve de acuerdo con los estándares establecidos internacionalmente o internamente.

El aseguramiento de la calidad aborda principalmente: un enfoque de gestión de la calidad, métricas del software, verificación y validación a lo largo del ciclo de vida del software, incluyendo pruebas y procesos de revisión y auditorías, gestión de configuración del software, el control de la documentación del software y un procedimiento que asegure los ajustes a los estándares en el proceso de desarrollo de software siempre que esto sea posible.

1.3.3.1 Actividades del aseguramiento de la calidad del software

Las actividades fundamentales que realiza o facilita el grupo de SQA son: (Navarro, 2002)

- **Planificación de la calidad:** Consiste en seleccionar, clasificar y ponderar las propiedades de calidad que se van a establecer como requisitos, con respecto al producto y con respecto al proceso. Se elegirán también los mecanismos de control de calidad a utilizar para medir y evaluar estas características, además de determinar las metas a alcanzar.
- **Supervisión de la calidad:** Consiste en supervisar y corregir, si es necesario, el trabajo que se está realizando (según los resultados obtenidos en las actividades de control de calidad), con el objetivo de llegar a satisfacer los requisitos establecidos.
- **Construcción de la calidad:** Actividades constructivas son aquellas que sirven para “construir” la calidad, es decir, son actividades preventivas cuyo objetivo es evitar la introducción de errores mediante la puesta en práctica de ciertos principios, métodos, formalismos y herramientas.

De forma general estas actividades se resumen en: un enfoque de gestión de la calidad; tecnología de ingeniería de software efectiva (métodos y herramientas); revisiones técnicas formales (RTF) que se aplican durante el proceso del software; una estrategia de prueba; el control de la documentación del software y de los cambios realizados; un procedimiento que se ajuste a los estándares de desarrollo del software cuando sea posible y por último un mecanismo de medición y de generación de informes. (PRESSMAN, 2002)

La autora de este trabajo considera que estas actividades constituyen el soporte del aseguramiento y que todas tienen importancia pues cada una forma parte de una actividad mayor que es garantizar la calidad del producto o software.

1.4 Verificación y Validación de requisitos. Definiciones

Como se había dicho anteriormente para asegurar la calidad del software se realizan un grupo de actividades, de estas, las que se enfocan en garantizar que son realizadas las funciones correctas se agrupan como actividades de Validación, las que van dirigidas a garantizar el funcionamiento correcto o fiable de estas funciones se agrupan como actividades de Verificación. (TIAN, 2005)

1.4.1 Verificación y Validación

La norma 610.12-1990 entre sus terminologías para la Ingeniería de Software trata la V&V de Requisitos como: El proceso de determinar si los requisitos para un sistema o componente son completos y correctos, los productos de cada fase de desarrollo cumplen con los requisitos o condiciones impuestas por la fase anterior, y el sistema o componente final cumple con los requisitos especificados. (IEEE, 1990) (PRESSMAN, 2002) citando a Barry Bohem plantea dos interrogantes que identifican claramente a la Verificación y a la Validación:

- Verificación: ¿Se está construyendo el producto correctamente?
- Validación: ¿Se está construyendo el producto correcto?

De la misma forma define que la **Verificación** se refiere al conjunto de actividades que aseguran que el software implementa correctamente una función específica, es decir, es el proceso de evaluar un sistema o componente para determinar si los productos de una fase satisfacen las condiciones impuestas al inicio de la misma está encaminada a que los productos de trabajo reflejen los requisitos especificados, asegurando que se está construyendo correctamente mientras que la **Validación** se refiere a un conjunto diferente de actividades que aseguran que el software construido se ajusta a los requisitos del cliente, es decir, es el proceso de evaluar un sistema o componente durante o al final del proceso de desarrollo para determinar si este satisface los requerimientos especificados. Demuestra que el producto cumplirá con el uso que tiene definido, asegurando que se está construyendo lo correcto.

El propósito de la **Verificación** es asegurar que el producto seleccionado cumple con los requisitos especificados, la misma involucra las siguientes actividades: la preparación de la verificación, la realización de la misma, e identificación de acciones correctivas. Incluye la verificación del producto final y del producto intermedio contra todos los requisitos seleccionados. Mientras que el propósito de la **Validación** es demostrar que un producto o componente cumple su uso intencional cuando se encuentra en el ambiente definido. Las actividades de validación pueden ser aplicadas a todos los aspectos del

producto en cualquiera de sus marcos intencionales, como puede ser el funcionamiento, el entrenamiento, la construcción, el mantenimiento y servicios de apoyo. La validación demuestra que el producto, realmente cumple con el propósito para el cual fue pensado, considerando que mediante esta actividad se comprueba si el producto propiamente refleja los requisitos especificados, asegurando que realmente se construyó el producto deseado por el cliente.

1.4.2 Objetivos del proceso de Verificación y Validación de requisitos

Gabriela Bermúdez y Marcelo Corona en su artículo Diseño de un estándar de Verificación y Validación para asegurar la calidad del software, exponen dos objetivos principales:

1. Verificar que los productos obtenidos en cada fase del ciclo de vida:
 - Cumplan con los requerimientos de la fase anterior, así como también, establezcan las bases apropiadas para iniciar la siguiente fase del ciclo de vida.
 - Satisfagan los estándares, prácticas y convenciones de la fase actual.
2. Validar que el producto final cumpla con los requerimientos del software establecidos. (BERMUDEZ and CORONA, 2002)

La (IEEE, 1998) también define los objetivos de este proceso, los cuales guardan relación con los primeros expuestos:

1. Facilitar la temprana detección y corrección de errores en el software.
2. Mejorar la gestión de los riesgos del proceso y el producto.
3. Apoyo a los procesos del ciclo de vida del software para asegurar el cumplimiento con la ejecución del programa, cronograma, y presupuesto.

En síntesis y de manera muy concreta la norma IEEE 1012-1998 indica que:

- El propósito de V&V de Requisitos es ayudar a la organización del desarrollo de software, para construir con calidad durante el ciclo de vida de software. (IEEE, 1998)

1.4.3 Ingeniería de requerimientos

La Ingeniería de Requerimientos (IR) cumple un papel primordial en el proceso de producción de software,

ya que se enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas. Una de las de las definiciones más generales de la IR es la siguiente:

Como se explicó anteriormente, se puede decir que la ingeniería de requerimientos sirve como una base sólida en el proceso de desarrollo de software, por lo que antes de pasar a tratar otros aspectos, es importante primero definir lo que es un requerimiento y cuáles serían las características deseables que deberían tener.

1.4.4 ¿Qué es un requisito o requerimiento?

Se presenta a continuación la definición existente en el glosario de la IEEE de lo que es un “Requerimiento”:

- “Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo”. (Std 610.12-1900, IEEE: 62)
- “Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal”.(Std 610.12-1900, IEEE: 62)

Analizando las definiciones anteriores, un requerimiento es una descripción de una condición o capacidad que debe cumplir un sistema, ya sea derivada de una necesidad de usuario identificada, o bien, estipulada en un contrato, estándar, especificación u otro documento formalmente impuesto al inicio del proceso. Estos presentan tres importantes propósitos:

1. Permiten a los desarrolladores entender cómo el cliente quiere que trabaje el sistema.
2. Especifican a los diseñadores la funcionalidad y las características que el sistema debe tener.
3. Especifican al grupo de prueba lo que deben demostrar para convencer al cliente que el sistema satisface sus necesidades.

1.4.5 Verificación y Validación en el Modelo de Capacidad y Madurez Integrado

CMMI (Modelo de Capacidad y Madurez Integrado), es un modelo de mejora del proceso de madurez para el desarrollo de productos y servicios. Se compone de las mejores prácticas que se ocupan de actividades de desarrollo y mantenimiento que cubren el ciclo de vida del producto desde su concepción hasta su entrega y mantenimiento. (INSTITUTE, 2006)

El CMMI contiene cinco niveles para evaluar que tan sofisticada es una organización en el establecimiento y apego a procesos estándares.



Figura 2: Niveles de CMMI

CMMI tiene 22 áreas de procesos clasificadas en cuatro categorías: Ingeniería, Gestión de Proyecto, Gestión de Procesos, Soporte, las cuales componen las dos representaciones que el modelo propone: la representación escalonada (*madurez*) donde se definen un conjunto de áreas por nivel de madurez y la representación continua (*capacidad*) en la cual se seleccionan las áreas a mejorar.

Entre estas áreas de proceso se encuentran la *Verificación* y *Validación*. La Validación demuestra que el producto cumplirá con el uso que tiene definido, mientras que la Verificación está encaminada a que los productos de trabajo reflejen los requisitos especificados. En otras palabras, la Verificación asegura que se está construyendo correctamente y la Validación asegura que se está construyendo lo correcto. (INSTITUTE, 2006)

Por la densidad de la explicación sobre cada una de estas áreas que el modelo propone solo se mostrarán algunos de los aspectos esenciales que trata (INSTITUTE, 2006):

Verificación

Mientras que el propósito de la *Verificación* es asegurar que el producto de trabajo seleccionado o el producto realizado son compatibles o responden a sus requisitos específicos. La *Verificación* es sin dudas un proceso incremental, porque está presente en todo el desarrollo del producto y productos de trabajo. (INSTITUTE, 2006)

Las *prácticas específicas* de esta área de proceso son realizadas cada una de la siguiente manera:

- ✓ La *Selección del Producto de Trabajo para la Verificación*, permite la identificación del producto de trabajo para ser verificado, los métodos que se usan para realizar la verificación y los requerimientos que deben ser satisfechos (o que debe cumplir) por cada producto de trabajo seleccionado.
- ✓ El *Establecimiento del Entorno de Verificación* permite la determinación del ambiente que se usará para llevar a cabo la verificación.
- ✓ El *Establecimiento de los Procedimientos y Criterios de Verificación* permiten el desarrollo de los procedimientos y criterios de verificación en línea con los productos de trabajo seleccionados, requerimientos, métodos, y las características del entorno de verificación.
- ✓ La *Ejecución de la Verificación* conduce la verificación acorde a los métodos permitidos, los procedimientos y criterios.

Validación

El propósito de la *Validación* es demostrar que un producto o componente de un producto cumple con su uso previsto cuando es puesto en práctica en el entorno previsto. El modelo incluye un conjunto de *prácticas específicas* para esta área de proceso:

- La práctica específica de *Seleccionar los productos para la validación* permite la identificación del producto o componente de producto para ser validado y los métodos que serán usados para llevar a cabo la validación.
- La práctica específica de *Establecer el entorno de validación* permite la determinación del ambiente que será usado para llevar a cabo la validación.

- La práctica específica de *Establecer los procedimientos y criterios de validación* permite el desarrollo de los procedimientos y criterios de validación que están alineados con las características de los productos seleccionados, con las restricciones del cliente en la validación, con los métodos y el entorno de validación.
- La práctica específica de *Ejecutar la validación* permite la ejecución de la validación de acuerdo con los métodos, procedimientos y criterios.

Ambas áreas de proceso se encuentran en el nivel 3 de madurez según la representación escalonada del modelo, algo que demuestra la importancia que le conceden dentro de la mejora de los procesos de desarrollo. Es imposible cumplir con lo que establecen modelos como CMMI para la Verificación y la Validación de Requisitos, si no se tiene en cuenta que el éxito en la implementación de estos procesos depende en gran medida de una correcta planificación para el entorno en el cual serán aplicados.

1.5 Métodos de Verificación y Validación de requisitos

Para dar cumplimiento a los objetivos de la Verificación y la Validación de Requisitos se utilizan los métodos o técnicas de control, que se dividen en dos grandes grupos: los *controles estáticos* y los *controles dinámicos*.

Los **métodos estáticos** tienen que ver con el análisis y control de las representaciones del sistema, es decir, de los diferentes modelos construidos durante el proceso de desarrollo de software, tales como documentos de requerimientos, diagramas de análisis y diseño, y código fuente. En esta categoría se encuentran las revisiones técnicas formales o inspecciones de programas, y herramientas de análisis estático, muchas de ellas provistas hoy en día como parte del compilador.

Los **métodos dinámicos**, también conocidos como pruebas, se basan en ejercitar una implementación. Por lo tanto, sólo pueden ser aplicados si existe una versión operativa o ejecutable del producto.

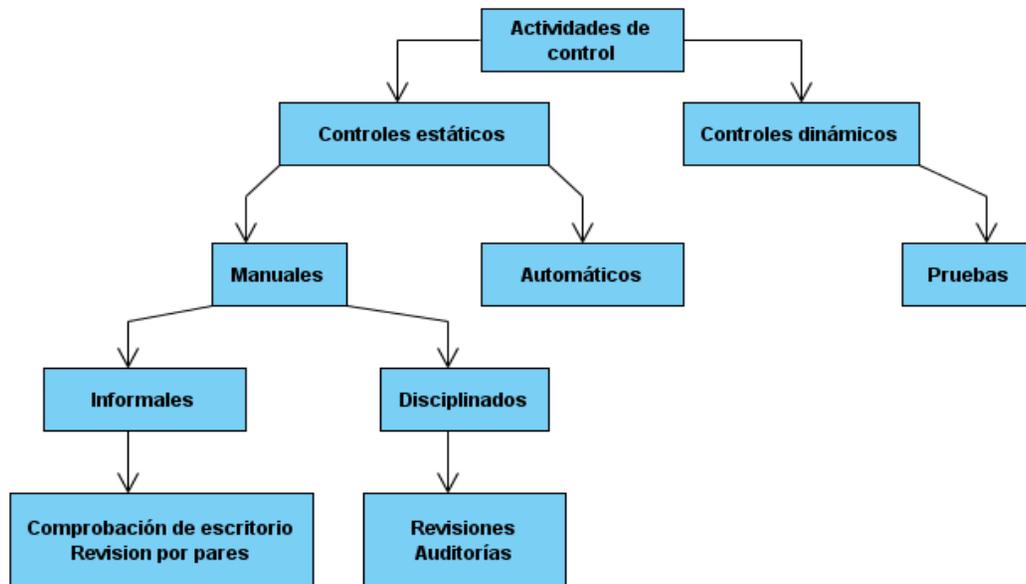


Figura 3. Actividades de control. Fuente: (ANTONIO, 1999)

1.5.1 Controles estáticos manuales informales

Estas actividades las realizan los propios autores de los objetos a comprobar, o personas de su misma categoría y ocupación.

La **comprobación de escritorio** (desk checking por su terminología en inglés) consiste en examinar a mano e individualmente el objeto que se acaba de desarrollar. Es el método tradicional para analizar un programa. Se debe aplicar a los requisitos, especificaciones de diseño y código según se van desarrollando. Es más efectivo si se hace intercambiando el objeto a examinar con otro compañero. (ANTONIO, 1999)

La **revisión por pares o iguales (peer review)** consiste en la revisión del código de un programador por otros programadores (sus pares). Se puede poner en práctica creando un panel que se encarga de revisar periódicamente muestras de código. (ANTONIO, 1999).

1.5.2 Controles estáticos manuales disciplinados

La misión principal de este tipo de control es que el control de la calidad no recaiga solo sobre el propio desarrollador.

1.5.2.1 Auditorías

La norma IEEE 1028-1997 plantea que la auditoría es el examen independiente de un producto de trabajo o grupo de productos de trabajo para evaluar el cumplimiento con especificaciones, estándares, acuerdos contractuales u otro criterio. Las auditorías constituyen la forma que tienen los auditores de software de verificar si se cumplen las disposiciones generales de calidad dictadas por el equipo SQA del proyecto y si se llevan a cabo todas las actividades de calidad necesarias para producir un software de buena calidad, con todos los artefactos necesarios para su buen entendimiento, y sobre todo si los estándares definidos en el proyecto son cumplidos eficazmente.

1.5.2.2 Revisiones técnicas

Una revisión técnica es una evaluación sistemática de un producto de software por un equipo de personal calificado, que examina la idoneidad del producto de software para su uso e identifica discrepancias con las especificaciones y estándares. Las revisiones técnicas también pueden proporcionar recomendaciones de alternativas y el examen de las distintas alternativas. (IEEE, 1997)

Existen varios tipos de revisiones, entre los cuales se encuentran las inspecciones, que son las revisiones que se hacen normalmente en un proyecto para verificar que todo está en orden, walk-through (de recorrido) y las revisiones técnicas formales, las que se especifican a continuación.

Inspecciones

Una inspección es un examen visual de un producto de software para detectar e identificar anomalías, incluyendo errores y desviaciones de los estándares y especificaciones. Las inspecciones son exámenes por pares dirigidos por facilitadores imparciales que están capacitados en técnicas de inspección. (IEEE, 1997)

Revisiones Técnicas Formales (RTF)

Una revisión técnica formal es el filtro más efectivo desde el punto de vista de garantía de calidad. Llevada a cabo por ingenieros del software (y otros), la RTF es para ellos un medio efectivo para mejorar la calidad del software. Su objetivo primario es encontrar errores durante el proceso, de forma que no se conviertan en defectos después de la entrega del software y su beneficio es el descubrimiento de errores al principio para que no se propaguen al paso siguiente del proceso del software. (PRESSMAN, 2002)

La RTF es realmente una clase de revisión que incluye recorridos, inspecciones, revisiones cíclicas y otro pequeño grupo de evaluaciones técnicas del software. Cada RTF se centra en una parte específica (y pequeña) del software total. Por ejemplo, en lugar de intentar revisar un diseño completo, se hacen inspecciones para cada módulo (componente) o pequeño grupo de módulos. Al limitar el centro de atención de la RTF, la probabilidad de descubrir errores es mayor.

1.5.3 Controles dinámicos. Pruebas

Pruebas

Para determinar el nivel de calidad de un sistema software se deben efectuar medidas o pruebas que permitan comprobar el grado de cumplimiento respecto de las especificaciones iniciales del sistema.

En la cadena de valor del desarrollo de un software específico, el proceso de prueba es clave a la hora de detectar errores o fallas. Conceptos como estabilidad, escalabilidad, eficiencia y seguridad se relacionan a la calidad de un producto bien desarrollado. Las aplicaciones de software han crecido en complejidad y tamaño, y por consiguiente también en costos. Hoy en día es crucial verificar y evaluar la calidad de lo construido de modo de minimizar el costo de su reparación. (FOUNDATION, 2008)

1.5.3.1 Objetivos de las pruebas

Citando a Glen Myers, (PRESSMAN, 2002) plantea una serie de normas que al decir del propio autor, pueden servir acertadamente como objetivos de las pruebas:

- La prueba es el proceso de ejecución de un programa con la intención de descubrir un error.
- Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.

1.5.3.2 Métodos o técnicas que se utilizan para realizar las pruebas

El software debe probarse desde dos perspectivas diferentes:

1. La lógica interna del programa se comprueba utilizando técnicas de diseño de casos de prueba de *caja blanca*.
2. Los requisitos del software se comprueban usando técnicas de diseño de casos de prueba de *caja negra*.

Prueba de caja blanca

Las pruebas de caja blanca se basan en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que examinen que están correctas todas las condiciones y/o bucles para determinar si el estado real coincide con el esperado o afirmado.

Las pruebas de caja blanca son consideradas uno de los tipos de pruebas más importantes que se le aplica al software, ya que consiguen una considerable disminución de los errores del sistema logrando que el software sea más confiable.

Prueba de caja negra

Las pruebas de caja negra se centran en lo que se espera de los módulos, es decir, intentan encontrar casos en que los módulos no se atienen a su especificación por ello se denominan pruebas funcionales. (MAÑAS, 1994). Se refiere a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Varios autores caracterizan este método de pruebas en tres aspectos fundamentales:

1. Los métodos de pruebas de caja negra se centran en los requisitos funcionales del software.
2. Con estas pruebas se obtienen conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa.
3. Las pruebas de caja negra no son una alternativa a las de caja blanca, sino que son un complemento para tratar de descubrir otros errores.

Según Pressman estas pruebas permiten encontrar: (PRESSMAN, 2002)

- Funciones incorrectas o ausentes.
- Errores de interfaz y en estructuras de datos o en accesos a las bases de datos externas.
- Errores de rendimiento.

Las pruebas de caja negra son las más comunes, ya que se basan en probar la funcionalidad y eso siempre se hace, ya sea por los desarrolladores o revisores. En un proceso más formal de pruebas son utilizados los Casos de pruebas que indican en modo en que debe comportarse el software ante una entrada de datos.

1.5.3.3 Niveles de pruebas

Alfredo Weitzenfeld en su libro *Ingeniería de software orientada a objetos con Java e Internet* argumenta que existen tres niveles principales para aplicar las diferentes técnicas de pruebas (WEITZENFELD, 2006):

- *Prueba de unidad:* Mediante esta prueba solo una unidad es probada como tal, como una clase, un paquete de servicio o un subsistema.
- *Prueba de integración:* En ella se verifica que las unidades trabajen juntas correctamente. Ambas pueden ser realizadas mediante casos de uso de pruebas, los cuales pueden ser aplicados a clases, paquetes de servicio, subsistema o el sistema completo.
- *Prueba de sistema:* Verifica el sistema completo o su aplicación como tal. Se toma el punto de vista del usuario final y los casos de uso de prueba ejecutan acciones típicas del usuario.

1.5.3.4 Herramientas utilizadas en las pruebas

Durante los procesos de auditorías, revisiones y pruebas es necesario utilizar herramientas y técnicas que agilicen el trabajo, con el objetivo de garantizar confiabilidad y mantener la integridad de los datos. En el proyecto se utilizarán las siguientes herramientas y técnicas para la realización de pruebas y auditorías.

Redmine

Es una herramienta de gestión de proyectos de software con interfaz web. Una vez instalada, el administrador da de alta los proyectos a través de la interfaz, puede dar de alta a los desarrolladores y jefes de proyecto (o pueden darse de alta a través de la interfaz web). Es una forma de agilizar la gestión de las no conformidades y verificar el tiempo que se necesita para resolver la misma, brindando la posibilidad de planificar y dar seguimiento a cada tarea que realiza el equipo. Redmine brinda más posibilidades, como:

- Envío automático de e-mail a los desarrolladores cada vez que se les asigna una tarea.

- Posibilidad de subir ficheros y documentos y de definir nuevos tipos de tareas y errores, con campos personalizados, todo ello fácilmente a través de la interfaz web.
- Se puede ver a través de Redmine los cambios en el repositorio.

TortoiseSVN

Subversion es un software de sistema de control de versiones. Es software libre bajo una licencia de tipo Apache/BSD y se le conoce también como SVN por ser ese el nombre de la herramienta de línea de comandos. Permite recuperar versiones antiguas de ficheros y examinar la historia de cuándo y cómo cambiaron los datos, y quién hizo el cambio. Esta herramienta es usada en la mayoría de los proyectos de la Universidad como herramienta para el control de versiones, es fácil de usar y amigable por sus iconos e interfaces personalizados, permite llevar un control de las versiones de los artefactos en un espacio determinado de tiempo.

JMeter

Apache JMeter es una herramienta de carga diseñada para realizar Pruebas de rendimiento y Pruebas funcionales sobre aplicaciones web. Originalmente fue diseñada para realizar pruebas de estrés sobre aplicaciones web (pruebas web clásicas) y con la misma se pueden atacar aquellos puntos que se quieran probar con respecto a concurrencia de usuarios y velocidad de carga en un sitio. En el proyecto ERP-Cuba esta herramienta será utilizada para realizar pruebas de estrés a los sistemas que en él se desarrollen.

1.6 Técnicas para la validación de los requerimientos de software

Las técnicas de validación de los requisitos se realizan con el fin de examinarlos para asegurarse de que definen el sistema adecuado. Dichas técnicas tienen gran importancia ya que permiten detectar los errores de forma temprana, con el fin de no conducir a resultados inesperados evitando provocar gastos excesivos y grandes pérdidas de tiempo. Entre las técnicas aplicadas en el proyecto ERP para validar requisitos se encuentra el Prototipo de Interfaz de usuario, la Lista de Chequeo y el Diseño de Casos de Prueba por la facilidad de su uso.

- **Prototipo de interfaz de usuario:** Esta técnica de validación se les muestra a los usuarios para que ellos obtengan una idea más precisa del producto final, es decir, en un diseño en menor porción que verificará que las especificaciones han sido realizadas de acuerdo con los requisitos

del sistema. En un primer momento estos prototipos no tienen que obligatoriamente ser los finales pues pueden surgir nuevas propuestas, que se llevan a escoger según el más usable, debido a que este prototipo muestra el comportamiento externo del sistema. (María José Escalona, 2002)

➤ ***Lista de chequeo***

Formulario de preguntas, las cuales dependen del objetivo para el cual son usadas. Es fácil de aplicar, resumir y comparar. Proporciona un apoyo mayor mediante preguntas que los probadores deben responder mientras leen el artefacto. Esta técnica proporciona listas que ayudan al probador a saber qué tipo de faltas buscar.

➤ ***Diseño de Casos de Prueba***

Conjunto de entradas de pruebas, condiciones de ejecución y resultados esperados desarrollados para cumplir un objetivo en particular o una función esperada. Siempre es ejecutada como una unidad, desde el comienzo hasta el final. Debe verificar si el producto satisface los requerimientos del usuario, tal y como se describe en las especificaciones de los requerimientos, así como también, si el producto se comporta como se desea, tal y como se describe en las especificaciones funcionales del diseño.

Para realizar la propuesta del Plan de procesos de Verificación y Validación de Requisitos, se hizo un estudio de estas técnicas, y se proponen actividades que estén vinculadas a los objetivos y metas de estas técnicas de validación.

1.7 Plan de Verificación y Validación de requisitos

Actualmente suele referirse como plan a un conjunto de pasos a realizar, para conseguir un objetivo el cual conciba las actividades a realizar para guiar el aseguramiento de la calidad. Mediante un plan, se recoge el cronograma de todas las actividades, así como sus responsables, recursos que facilitan la correcta ejecución de las mismas. La planificación constituye uno de los requisitos fundamentales para el desarrollo eficiente y la calidad de un producto de software.

El propósito de planificar la V&V de Requisitos es asegurar que cada recurso, rol y responsabilidad estén claramente asignado, el plan V&V de Requisitos que se obtiene de este proceso documenta y describe los diferentes recursos, sus roles y actividades, así como las técnicas y herramientas a ser utilizadas. El plan también dirige la gestión, la documentación, políticas y procedimientos de las actividades V&V de

Requisitos y su interacción, así como el reporte de defectos y requerimientos de documentación. (ABRAN and MOORE, 2004)

La norma IEEE 1012-1998 es un estándar que abarca todos los procesos del ciclo de vida del software, incluyendo la Adquisición, Suministro, Desarrollo, Operación y Mantenimiento. Planificar la Verificación y Validación de Requisitos ajustándose a este estándar implica la comprensión obligatoria de todos los planteamientos y conceptos que en él se reflejan:

- Mediante el proceso de V&V de Requisitos del software se determina si los productos del desarrollo de una determinada actividad se ajustan a los requisitos de esa actividad, y si el software cumple con el uso previsto y las necesidades de los usuarios. La determinación incluye la evaluación, el análisis, revisión, inspección, y las pruebas a los procesos y productos del software. La V&V de Requisitos se realiza en paralelo con el desarrollo de software, no en la conclusión de este.
- Los resultados de V&V de Requisitos permiten a la organización de desarrollo de software modificar los productos de manera oportuna y reducir el costo total del proyecto y el impacto en el cronograma.(IEEE, 1998)

La siguiente figura muestra el marco de trabajo de V&V definido por el estándar, con la jerarquía de Procesos V&V → Actividades V&V → Tareas V&V.

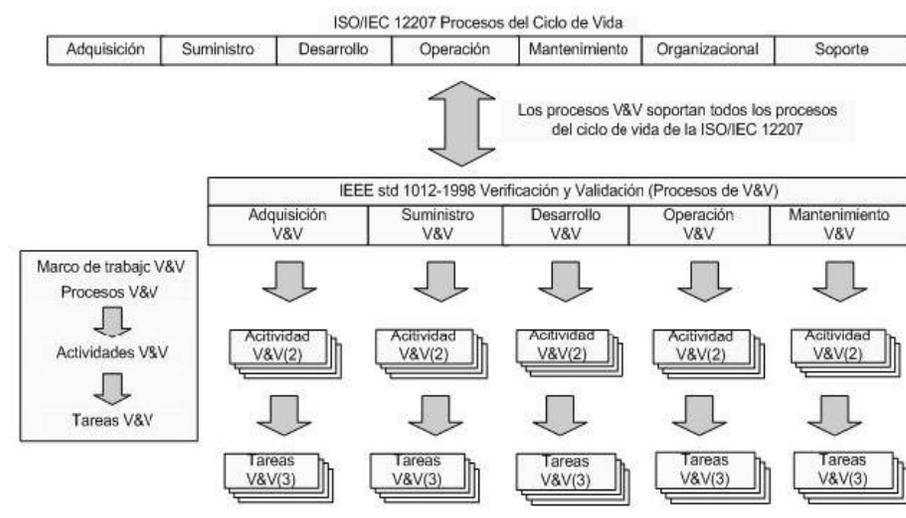


Figura 4. Marco de Trabajo V&V. Jerarquía de procesos, actividades y tareas. Fuente:(IEEE, 1998)

La IEEE 1012-1998 facilita a quien persigue el objetivo de planificar la Verificación y la Validación de Requisitos de un software:

- El establecimiento de una base teórica completa para la comprensión de todo lo que concierne a la V&V de Requisitos con conceptos claros y aplicables de ambos procesos.
- Un esquema de niveles de integridad que puede ser adoptado para definir a partir de la criticidad, el nivel de integridad de cualquier software.
- La definición de los contenidos requeridos para un Plan de Verificación y Validación de Requisitos de un software, soportado por procesos V&V, donde cada proceso está conformado por actividades V&V y estas a su vez por tareas V&V a ejecutar durante todo el ciclo de vida.
- La estructura para un plan de este tipo con la explicación detallada de cada una de sus secciones.

Teniendo en cuenta esto, y luego del análisis exhaustivo que permitió conocer no solo los aspectos reflejados anteriormente, sino otros de importante significación para este estudio, se decidió adoptar la IEEE 1012-1998 *Estándar para la Verificación y Validación de Software* para cumplir el objetivo general de la investigación, por lo que la propuesta de solución estará basada en el mencionado estándar

1.9. Conclusiones parciales

Los procesos de Verificación y Validación de Requisitos son claves para asegurar la calidad de un producto informático, algo que se puede concluir del estudio efectuado sobre estos en el capítulo y para darle cumplimiento a sus objetivos deben ser planificados y ejecutados durante todo el ciclo de vida del software. Después de haber realizado un exhaustivo y minucioso análisis y estudios de las normas y estándares actuales relacionados con el trabajo a realizar se determinó que la propuesta del Plan de Verificación y Validación de Requisitos para el proyecto ERP-Cuba estará sustentada por la norma IEEE 1012-1998. De esta se obtuvo un cúmulo significativo de información relacionada con la puesta en práctica de los procesos de Verificación y Validación de Requisitos a lo largo de todo el ciclo de vida del software, así como su planificación. Esto permitió identificar nuevas actividades y tareas V&V a aplicar en el ERP-Cuba y determinar los métodos que se pueden implementar en el proyecto para dar cumplimiento a cada tarea, analizando la importancia de utilizar tanto mecanismos de control estático, como dinámicos, y no solamente pruebas al software.

CAPÍTULO 2. PROPUESTA DEL PLAN DE VERIFICACIÓN Y VALIDACIÓN DE REQUISITOS

2.1 Introducción

El presente capítulo describe la propuesta del Plan de V&V de Requisitos a utilizar en el proyecto ERP específicamente en la línea de Capital Humano comenzando por una caracterización general de dicho proyecto para ayudar a la comprensión del entorno sobre el cual se trabaja en la investigación. Luego se explican los aspectos a tener en cuenta para adaptar la norma IEEE 1012-1998 a las características particulares del proyecto con el que se interactúa. Seguidamente, se muestra todo lo referido al Plan de Verificación y Validación de Requisitos, su estructura, actividades y tareas, además de los métodos y procedimientos que en él se aplican.

2.2 Descripción del proyecto ERP- Cuba

ERP-Cuba es un proyecto productivo creado a partir de la necesidad del país de contar con un sistema capaz de garantizar que las entidades puedan compartir la información de sus procesos en el menor tiempo posible por lo que se decide acometer el desarrollo de un sistema enfocado a planificar los recursos empresariales, apoyándose en un equipo multidisciplinario compuesto por Especialistas de diferentes entidades del país: la Universidad de las Ciencias Informáticas y la atención directa de los Organismos de la Administración Central del Estado. Tiene como objetivo principal brindar una solución nacional que permita, a diferencia de otros productos similares, la gestión integral de las entidades presupuestadas y empresariales, basada en los principios de independencia tecnológica y con funcionalidades generales de los procesos y las particularidades de la economía cubana. (Del Toro Ríos & González Brito, 2008). El proyecto ERP-Cuba se divide en 8 Líneas de desarrollo las cuales son: Estructura y Composición, Configuración, Auditoría, Capital Humano, Costos y Procesos, Contabilidad y Finanzas, Logística y Planificación. Como se había dicho anteriormente la investigación se centrará en la línea Capital Humano la cual está compuesta por módulos como Configuración, Organización del trabajo, Selección e integración, Administración de Capital Humano, Competencias Laborales, Estimulación Moral y Material, Evaluación de desempeño, Capacitación y desarrollo, Seguridad y Salud en el trabajo, Comunicación Institucional, Autocontrol, Sistema de trabajo con los cuadros y su reserva. Esta línea cuenta con tres respectivos componentes tales como: Organización del trabajo el cual tiene incluido, Puesto de trabajo y Régimen de trabajo, la Selección e integración aquí se tiene a Persona y por último la

Administración de Capital Humano con Trabajador y Pagos adicionales. La distribución de roles es realizada como se muestra en la Figura 5. Cada rol tiene definido el flujo de trabajo que debe seguir para realizar sus responsabilidades. La preparación previa al desarrollo permitió que cada programador domine en detalle la arquitectura definida para el proyecto, algo muy importante a la hora de implementar las funcionalidades de cada módulo. Existe una comunicación constante entre el Arquitecto del sistema y los programadores de su equipo de desarrollo, de igual manera con el equipo de arquitectura para la toma de decisiones que impliquen cambios arquitectónicos. Los analistas son un eslabón imprescindible para el completo entendimiento del negocio por parte del equipo de desarrollo encargado de implementar las funcionalidades relativas a este.

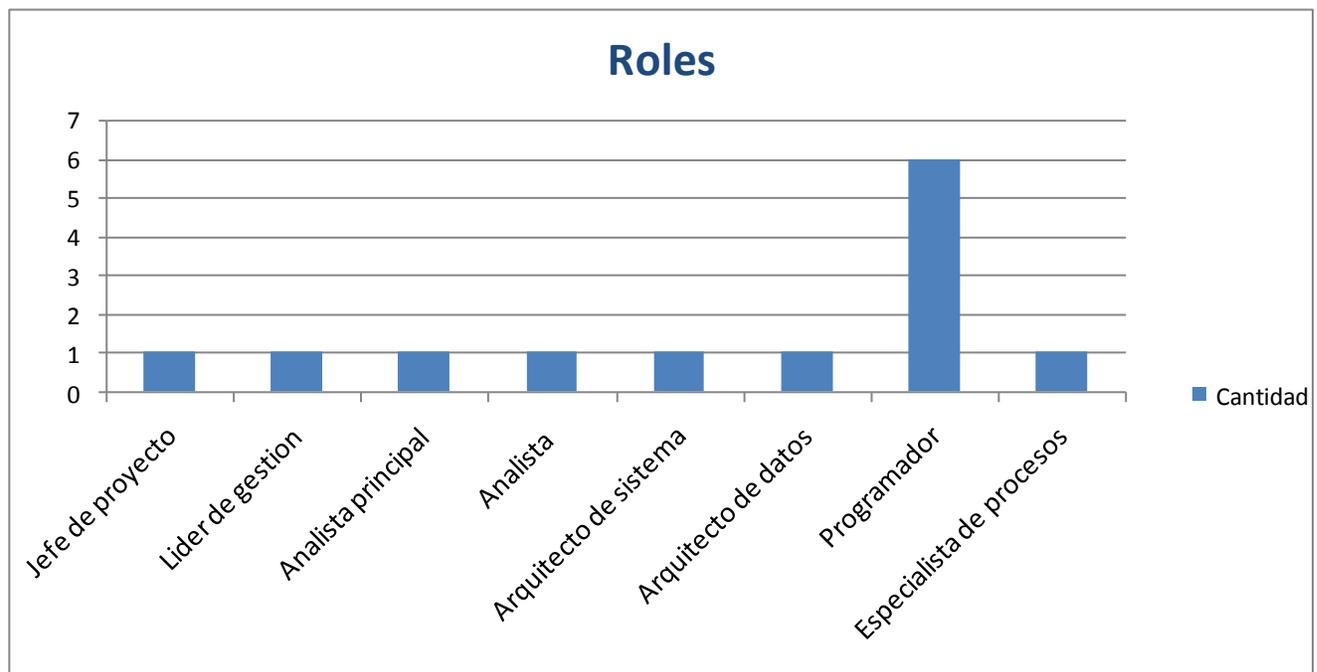


Figura 5. Distribución de roles del proyecto ERP (Capital Humano)

2.3 Guía de adopción del estándar

Para adaptar el estándar IEEE 1012-1998 a las características particulares del proyecto ERP, se tuvieron en cuenta 2 aspectos fundamentales:

1. El nivel de integridad del software: es un parámetro que mide la criticidad del software y que después de definido será el punto fundamental a tener en cuenta para tomar las decisiones precisas sobre los próximos tres aspectos.
2. Los criterios que se emplearán en las tareas V&V para establecer los parámetros mínimos de corrección, completitud, consistencia, exactitud, legibilidad y comprobabilidad.

2.3.1 Nivel de integridad del software

Para determinar el nivel de integridad del ERP se realizó un análisis de la criticidad del software, que implica directamente el análisis de los riesgos y daños que existen ante el fallo del sistema o de algunas de sus funciones para cuantificar su posible impacto, y de esta manera determinar el nivel de integridad, teniendo en cuenta las posibilidades de mitigar estos riesgos y daños. Para este análisis se adoptó el esquema de niveles de integridad que propone la IEEE 1012-1998.

Tabla 2. Esquema de niveles de integridad

Nivel	Dimensión del daño por fallo del software	Mitigación posible
4	<ul style="list-style-type: none">- Pérdidas de vidas humanas- Pérdida de la seguridad del sistema- Graves pérdidas económicas o sociales	No es posible mitigar los daños producidos
3	<ul style="list-style-type: none">- El sistema no completa el objetivo que tiene previsto.- Graves pérdidas económicas o sociales	Es posible una mitigación parcial de los daños
2	<ul style="list-style-type: none">- El sistema incumple parcialmente con el objetivo que tiene previsto.- Pérdidas sociales o económicas importantes	Se pueden mitigar los daños producidos
1	<ul style="list-style-type: none">- Una determinada función no se realiza.- Consecuencias mínimas	No es necesario mitigar los daños.

El análisis de dicha integridad se efectuó en una reunión de trabajo con el analista principal y el Jefe de proyecto, los cuales son especialistas en el negocio. Se examinó cada módulo y componente según el objetivo y el impacto que tiene dentro del sistema y se llegó a la siguiente conclusión:

- *Aunque los módulos y componentes de la línea Capital Humano tienen diferentes objetivos y el impacto de cada uno para el sistema es distinto, tomando como base el esquema propuesto por la norma IEEE 1012-1998, pueden clasificarse todos en el nivel 2 de integridad.*

Tomando esto en cuenta, todos los aspectos que dependen del nivel de integridad del software, se tratarán atendiendo al nivel 2, y sobre esta base se organizarán los esfuerzos V&V.

2.3.4 Criterios que se emplearán en las tareas

Las tareas del plan están enfocadas a medir parámetros como aceptación de los requisitos, consistencia, correctitud, completitud, exactitud, legibilidad o comprobabilidad de un producto de trabajo determinado. Como conceptos globales estos términos se refieren a:

- ✓ *Aceptación de los requisitos:* Establecer los criterios para aprobar las descripciones de requisitos de software.
- ✓ *Consistencia:* El grado de uniformidad, estandarización, y libertad de contradicciones entre la documentación o parte de un sistema o componente.
- ✓ *Correctitud:* El grado en que un sistema o componente está libre de fallas en su especificación, diseño e implementación.
- ✓ *Completitud:* Nivel o grado en que algo está completo.
- ✓ *Consistencia:* Determinar si la Especificación de requisitos es consistente.
- ✓ *Legibilidad:* Calidad de la escritura que hace que sea fácil de leer y entender
- ✓ *Comprobabilidad:* El grado en que un sistema o componente facilita el establecimiento de criterios de prueba y la ejecución de las pruebas para determinar si esos criterios han sido satisfechos.

En el plan se detallan los criterios específicos a evaluar en cada parámetro dentro de las tareas. Asegurar que un producto de trabajo satisface esos criterios, es un aporte significativo (desde ese nivel) a la calidad con que se desarrolla el producto.

2.4 El Plan de Verificación y Validación de requisitos para el proyecto ERP (Capital Humano)

A partir de las características del proyecto ERP, con la base teórica estudiada, y con la definición de adoptar la norma IEEE 1012-1998, se crea el Plan de Verificación y Validación de Requisitos.

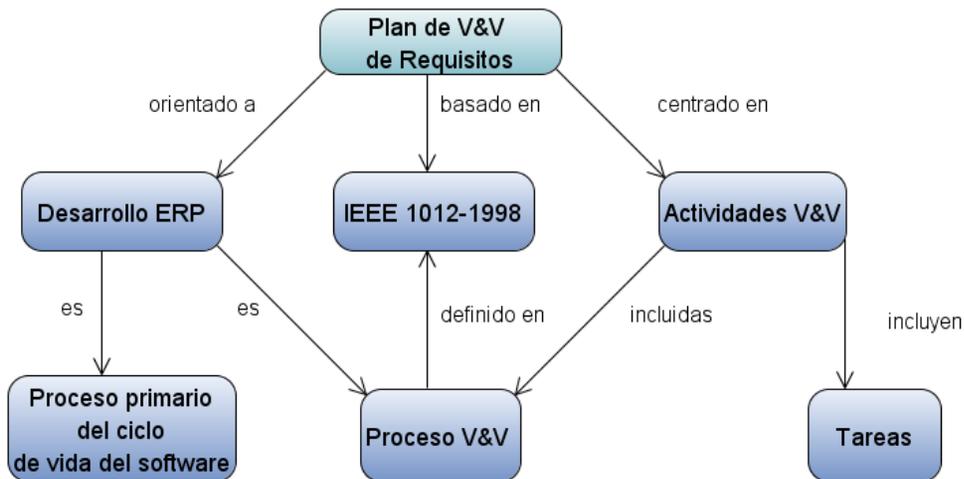


Figura 6. Vista general del Plan V&V de Requisitos del ERP

2.4.1 Estructura

La estructura está sostenida por la que propone la norma adoptada, con las variaciones necesarias para una mejor comprensión de su contenido y facilidad de aplicación en el entorno del proyecto.

Tabla 3. Estructura del Plan V&V del ERP

Plan V&V de Requisitos para el ERP	
1.	Propósito
2.	Documentos referenciados
3.	Definiciones y Abreviaturas
4.	Vista General V&V de Requisitos
4.1	Cronograma general
4.2	Esquemas de niveles de integridad
4.3	Recursos
4.4	Responsabilidades
5.	Procesos V&V
5.1	Proceso de Desarrollo
5.1.1	Actividad: Estudio preliminar V&V
5.1.2	Actividad: Modelado de negocio V&V
5.1.3	Actividad: Requisitos V&V
5.1.4	Actividad: Diseño V&V
5.1.5	Actividad: Implementación V&V
5.1.6	Actividad: Pruebas internas V&V
5.1.7	Actividad: Pruebas de liberación V&V
5.1.8	Actividad: Despliegue V&V
5.1.9	Actividad: Soporte V&V

A continuación se detallan sus elementos principales:

2.4.1.1 Propósito

En el propósito se especifican los objetivos que se persiguen con la planificación de la Verificación y la Validación de Requisitos para el proyecto, puntualizando el (los) procesos del ciclo del vida del software para el cual se prepara el plan.

2.4.1.2. Documentos referenciados

Se especifican los documentos referenciados dentro del Plan V&V de Requisitos.

2.4.1.3. Definiciones y abreviaturas

Se definen todos los términos necesarios para una buena comprensión del plan además de las abreviaturas utilizadas dentro del mismo.

2.4.1.4. Cronograma general

Se propone un cronograma tipo (Figura 9), para lograr un plan más flexible y aplicable a las iteraciones del proyecto, sin restricciones a fechas exactas, que especifica los siguientes elementos:

- El momento dentro del desarrollo en que se ejecuta la actividad, ilustrando la correspondencia entre las actividades V&V y las del desarrollo del ERP.
- El orden de ejecución de las actividades.
- Las actividades que componen una iteración (Estudio preliminar V&V, Modelo de negocio V&V, Requisitos V&V, Diseño V&V, Implementación V&V, Pruebas internas V&V, Pruebas de liberación V&V, Despliegue V&V, Soporte V&V) en relación con las características del proyecto.

Los recursos humanos (roles) necesarios para cumplir una actividad no se precisan en el cronograma, estos son descritos para cada tarea en específico.

2.4.1.5. Esquema de niveles de integridad

Se muestra el esquema de niveles de integridad adoptado para determinar el nivel de cada módulo y componente del proyecto. (Tabla 2)

2.4.1.6. Recursos

En esta sección se hace un resumen de los recursos humanos (distribuidos por roles) y tecnológicos con los que cuenta el proyecto para la puesta en práctica del plan. Existen roles que aunque no forman parte del equipo de desarrollo, juegan un papel importante para la realización de algunas tareas y por eso se especifican, es el caso del *Especialista de procesos*.

Los recursos tecnológicos son básicamente computadoras asignadas a los miembros del proyecto y otros dispositivos como escáner de huellas y cámaras digitales que en este caso no se cuenta con ellos. Las

computadoras están distribuidas en un laboratorio asignados a la línea de Capital Humano de manera tal que cada integrante tiene su puesto y horario de trabajo, constituyendo de esta forma el recurso más importante para la realización de las tareas V&V.

2.4.1.7. Responsabilidades

La asignación de responsabilidades se hace teniendo en cuenta que el primer esfuerzo V&V lo realiza el desarrollador (analistas, diseñadores, documentadores, programadores, en dependencia de la actividad que se esté realizando), que debe garantizar la calidad del proceso que ejecuta y del producto que se obtiene del mismo. La descripción detallada de las funciones precisas de cada rol con fines V&V se realiza dentro de cada tarea. Un mismo rol puede asumir diferentes responsabilidades, dependiendo de la tarea en la cual participa.

2.4.1.8. Procesos V&V

Se especifican todos los procesos para los cuales se está planificando la Verificación y la Validación, describiendo las actividades y tareas que incluyen.

La norma IEEE 1012-1998 define cinco procesos de Verificación y Validación que cubren todo el ciclo de vida del software: Adquisición V&V, Suministro V&V, Desarrollo V&V, Operación V&V, Mantenimiento V&V.

- *Adquisición V&V:* Este proceso comienza con la definición de la necesidad de adquirir un software, un producto o servicio de software. Los esfuerzos V&V en este proceso están dirigidos a determinar el alcance inicial de la Verificación y Validación, planificar las interacciones entre el comprador y el suministrador, y revisar el diseño inicial de los requerimientos del software contenidos en la solicitud de oferta.
- *Suministro V&V:* El proceso es iniciado por cualquier decisión de responder a la solicitud de oferta del adquiridor o por firmar y acceder al contrato con este para suministrar el software, producto o servicio de software. Continúa con la determinación de los procedimientos y recursos necesarios para administrar el proyecto, incluyendo el plan del proyecto y su ejecución a través del desarrollo del software, producto o servicio de software para el adquiridor. Los esfuerzos V&V están dirigidos a verificar que el diseño inicial de los requerimientos del software contenidos en la solicitud de oferta y los requerimientos del contrato son consistentes y satisfacen las necesidades del usuario.

CAPÍTULO II. PROPUESTA DEL PLAN DE V&V DE REQUISITOS

- *Desarrollo V&V*: El proceso de desarrollo contiene las actividades y tareas del desarrollador. Se incluyen las actividades referentes a la modelación del negocio, análisis de requerimientos, diseño, codificación, pruebas, despliegue y soporte de los productos de software. Estas actividades están organizadas en: Estudio preliminar V&V, Modelo de negocio V&V, Requerimientos V&V, Diseño V&V, Implementación V&V, Pruebas internas V&V, Pruebas de liberación V&V, Despliegue V&V, y Soporte V&V.
- *Operación V&V*: Este proceso cubre la operación del producto de software y el soporte operacional para los usuarios.
- *Mantenimiento V&V*: El proceso es activado cuando el producto de software sufre modificaciones al código y la documentación asociada, causado por un problema o necesidad de mejora o adaptación.

Los procesos de Adquisición V&V y Suministro V&V no se encuentran en la propuesta, puesto que son etapas ya realizadas en el proyecto y no es necesario dedicar esfuerzos a ellos.

El plan contiene las actividades y tareas del Desarrollo V&V, que es el proceso más extenso de todo el ciclo de vida y al cual se dedican la mayoría de los esfuerzos.

Los procesos de Operación V&V y Mantenimiento V&V tampoco se incluyen en la propuesta, pero son objeto de estudio para su posterior planificación.

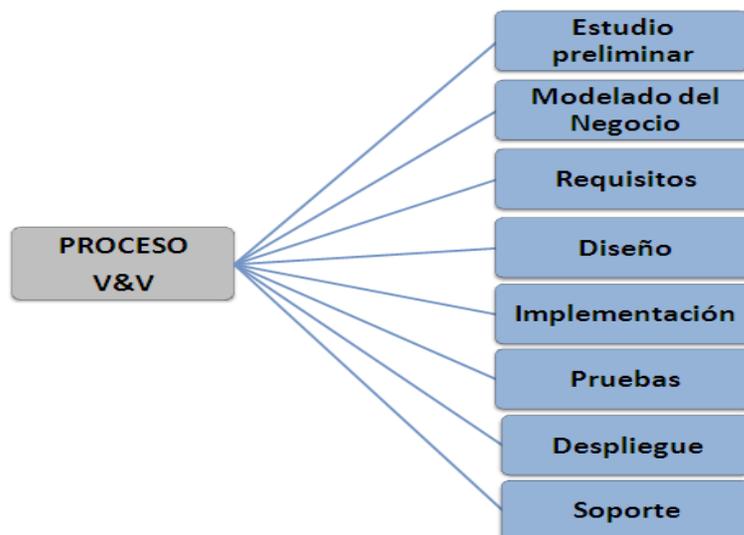


Figura 7. Proceso de desarrollo V&V

2.4.2 La actividad de Verificación y Validación de requisitos

La actividad es el elemento principal para organizar los esfuerzos V&V. Cada una se dirige a una etapa diferente dentro del desarrollo del software, constituyendo los puntos de relación directa con el entorno de trabajo que se evalúa, por ejemplo la correspondencia entre Implementación V&V y la fase de Implementación del ERP. Para la culminación exitosa de una actividad se requiere del cumplimiento de todas las tareas que la componen.

En función de los aspectos que se tuvieron en cuenta para la adopción de la norma, el Proceso de Desarrollo V&V del plan tendrá las mismas actividades que plantea la IEEE 1012-1998, ajustándolas a las particularidades del proyecto y redefiniendo los aspectos necesarios de cada una de ellas. De esta manera se describen:

Estudio preliminar

En la disciplina de Estudio Preliminar se llevan a cabo las actividades relacionadas con la planeación del proyecto a un alto nivel, la evaluación de la factibilidad del proyecto y la legalización del mismo. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.

Modelado del negocio

El Modelado del Negocio es la disciplina destinada a comprender los procesos de negocio de una organización. Se comprende cómo funciona el negocio que se desea automatizar para tener garantías de que el software desarrollado va a cumplir su propósito. La descripción del negocio se realiza guiada por procesos de negocio.

Requisitos

El esfuerzo principal en la disciplina de Requisitos es desarrollar un modelo del sistema que se va a construir. En la disciplina requisitos se identifican y describen los requisitos funcionales del sistema y los requisitos no funcionales.

Diseño

En la disciplina de Diseño se define la arquitectura del sistema y se realiza el diseño de cada componente del software para satisfacer los requisitos impuestos.

Implementación

En la implementación, a partir de los resultados del diseño se implementa el sistema en términos de componentes.

Pruebas internas

Durante esta disciplina el proyecto verifica el resultado de la implementación probando según cada construcción, incluyendo tanto las construcciones internas como intermedias, así como las versiones finales a ser liberadas. En esta disciplina se deben desarrollar artefactos de prueba como: Diseños de casos de prueba, Listas de chequeo y de ser posibles componentes de prueba ejecutables para automatizar las pruebas. Las revisiones al producto y al proceso de desarrollo de software se realizan como parte de esta disciplina. Las pruebas internas incluyen además pruebas funcionales.

Pruebas de liberación

Pruebas diseñadas y ejecutadas por Calidad UCI a todos los entregables de los proyectos antes de ser entregados al cliente.

Despliegue

Desarrollo de las pruebas piloto en el entorno real. Entrega de sistema bajo pruebas de liberación, entrega de código (si lo lleva), transferencia tecnológica (si lo lleva), escritura de manuales de usuario y documentación técnica afín, capacitación en el uso del sistema (si lo lleva).

Soporte

Desarrollo de las pruebas de regresión. Desmonte de la línea base y archivo del producto. Mantenimiento del producto.

2.4.3 La tarea de Verificación y Validación de requisitos

La tarea es el componente más pequeño del Plan V&V. Está orientada a evaluar un producto de trabajo determinado dentro del desarrollo. La estructura que se propone cumple con lo establecido en la IEEE 1012-1998 y se muestra en la siguiente figura:

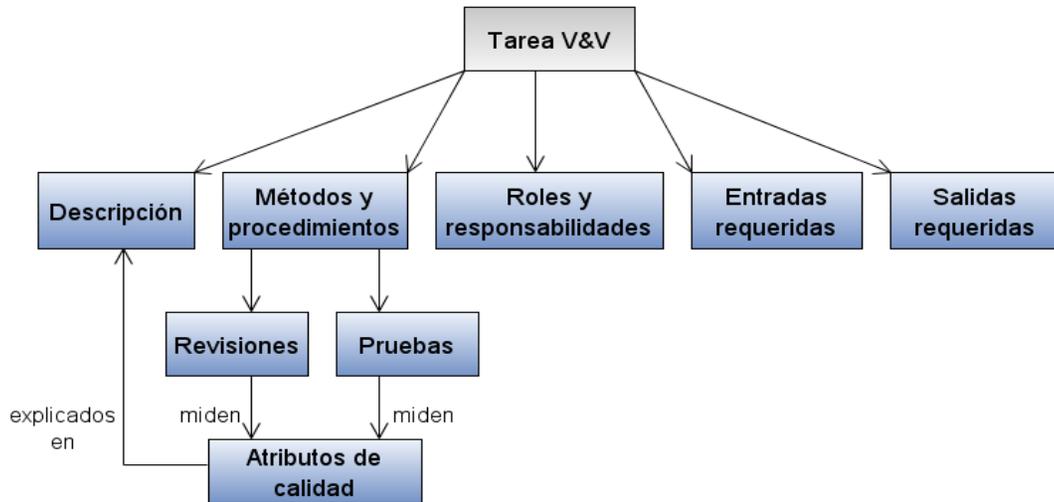


Figura 8. Estructura de la Tarea V&V

En la *descripción* se especifica claramente la manera en que se evaluará el producto de trabajo, atendiendo a los parámetros de corrección, completitud, consistencia, exactitud, legibilidad y comprobabilidad, para los cuales son definidos los criterios a tener en cuenta.

En los *métodos y procedimientos* se especifica el(los) método(s) que se utilizará(n) para darle cumplimiento a la tarea, así como el modo de aplicación (procedimiento) dentro del proyecto. Para la selección de los mismos se tuvieron en cuenta los conceptos que aporta la teoría estudiada, pero generalmente se aplican con algunas peculiaridades que derivan de las características y necesidades del ambiente productivo en el que se llevan a cabo.

Las *entradas requeridas* son aquellos artefactos necesarios para la ejecución de la tarea. Pueden constituir el objeto de revisión o prueba, o un material de apoyo vital para ello.

Las *salidas requeridas* son artefactos que se obtienen como resultado de la ejecución de la tarea. Las exigidas dentro del plan están en correspondencia con la norma IEEE 1012-1998, con las respectivas adaptaciones a la documentación generada en cada actividad del ERP

Se tienen artefactos como el Reporte de Tarea, donde se registra el resultado de la ejecución de cada tarea del plan, pues este constituye una evidencia fundamental de que se realizó, además de quién, cómo

y cuándo la ejecutó y de la cantidad de errores detectados. Este documento es fundamental para obtener datos estadísticos de la V&V y se realiza de forma automática a través de la herramienta Redmine. (Anexo C)

Para la selección de las tareas a incluir en el plan, se realizó un análisis de las tareas mínimas requeridas para el nivel de integridad 2 establecidas por la norma, a las cuales se le hicieron las variaciones requeridas para lograr una correcta adaptación a la realidad del ERP, teniendo en algunos casos que unificar tareas y en otros adicionar o quitar.

2.4.4 Métodos y procedimientos aplicados

La aplicación de los métodos y procedimientos en el plan está muy orientada a la singularidad con que se realizan las tareas del desarrollo ERP, insertadas en un cronograma muy estricto que deriva de importantes compromisos contractuales de obligatorio cumplimiento.

Las revisiones como método estático de V&V son aplicadas en todas sus variantes; se tiene la comprobación de escritorio realizada por el propio desarrollador, miembro del equipo de calidad, jefe de proyecto o cualquier integrante del equipo de desarrollo con la preparación necesaria para jugar el papel de revisor en el caso que se necesite. A pesar de esto las revisiones en el Centro se realizan orientadas por la propia Dirección General o por necesidades que el revisor líder de la Subdirección de Gestión de Calidad identifique y considere, por ello, necesario ejecutar este proceso, apoyándose en un equipo de revisores capacitado y acreditado para desempeñar este rol. También se hacen revisiones técnicas en equipos, ya sean reuniones orientadas a la revisión puramente o las que propicia el trabajo en conjunto de un determinado equipo del proyecto. Los walk-through guiados por analistas, diseñadores y probadores, son aplicados para la obtención y revisión de artefactos fundamentales en actividades como Modelado de Negocio V&V, Diseño V&V y Pruebas V&V. Los peer review que requieren del trabajo obligatorio de dos personas, también juegan un papel muy importante dentro del plan.

La prueba como método dinámico de V&V tiene un peso importante en el plan, a ella se dedica una actividad completa dentro del proceso de desarrollo, como lo plantea la IEEE 1012-1998. La generación y verificación del diseño, procedimientos y casos de prueba se hace conjuntamente a las actividades de Pruebas V&V, que es donde se ponen en práctica, y las pruebas unitarias se ejecutan en la actividad de Implementación V&V.

El plan incluye pruebas de unidad para la capa de acceso a datos y de negocio, integración, pruebas de sistema, pruebas de liberación y las pruebas de aceptación en estas últimas se comprueba que se satisfacen los requisitos pactados con el cliente, se describen como pruebas funcionales sobre el total de requisitos distribuidos por procesos del negocio. Para las pruebas de sistema se propone la realización de pruebas de funcionalidad, rendimiento, estrés y seguridad. Para la prueba de integración se comprueba el buen funcionamiento de los requisitos distribuidos entre los componentes de uno o varios subsistemas, la comunicación a través de interfaces y el acceso coherente a estructuras de datos globales.

2.5 Propuesta del Plan de Verificación y Validación de requisitos para el ERP

1. Propósito

Organizar y estandarizar los procesos de Verificación y Validación (V&V) en el Proyecto ERP, planificando las actividades y tareas V&V a realizar durante la fase de Desarrollo del ciclo de vida del software. Esta fase incluirá desde el estudio preliminar hasta la instalación y aceptación del producto final.

2. Documentos referenciados

No existen documentos referenciados.

3. Definiciones y Abreviaturas

DDR: Documento de Descripción de Requisitos

DNC: Documento de No Conformidades

DP: Documento de Pruebas definido para el proyecto

DTN: Descripción Textual del Negocio

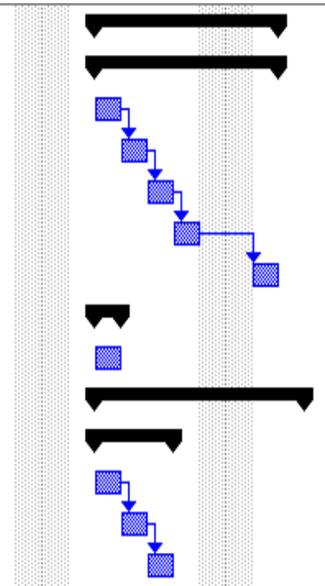
DV: Documento Visión

V&V: Verificación y Validación

4. Vista general de V&V

4.1 Cronograma general

Concepción
Estudio preliminar V&V
Realizar un estudio del arte.
Realizar un estudio técnico
Realizar un estudio económico
Realizar un estudio comercial
Decidir la ejecución del proyecto
Modelado del negocio V&V
Revisión de los procesos de negocios
Elaboración
Requerimientos V&V
Análisis de la trazabilidad
Evaluación de los requerimientos del software
Análisis de la interfaz usuario



Diseño V&V
Análisis de la trazabilidad.
Evaluación del diseño del software
Análisis de las interfaces.
Generación y verificación del Plan de Pruebas de Unidad V&V
Generación y verificación del Plan de Pruebas de Integración V&V
Generación y verificación del diseño de las pruebas V&V
Costrucción
Implementación V&V
Análisis de la trazabilidad.
Evaluación del código y su documentación
Generación y verificación de los Casos de Prueba V&V.
Generación y verificación de los Procedimientos de Prueba V&V
Ejecución y verificación de las pruebas unitarias

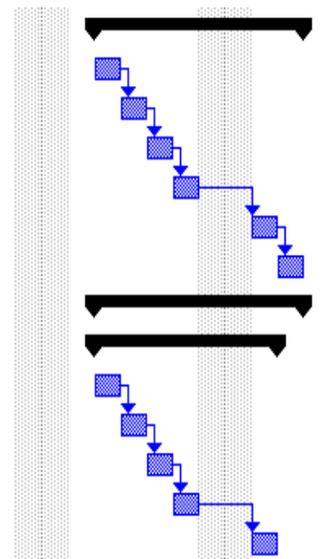




Figura 9. Cronograma tipo de las actividades del Plan de V&V

4.2 Esquema de niveles de integridad

En la tabla 2 se muestra el esquema de niveles de integridad adoptado para determinar el nivel de integridad del proyecto dichos niveles estarán asociados a cada módulo y componente del proyecto y se obtuvieron de la reunión realizada con los especialistas en el negocio.

4.3 Recursos

Los recursos humanos con que cuenta el proyecto para realizar las tareas V&V se muestran en la tabla que se presenta a continuación:

Tabla 4. Recursos humanos del proyecto ERP (Capital Humano)

Rol	Cantidad(Personas)
Jefe de proyecto	1
Planificador	1
Analista principal	1
Analista	1
Arquitecto de sistema	1
Arquitecto de datos	1
Programadores	6
Especialista de procesos	1
Total	13

Los recursos tecnológicos del proyecto se agrupan en tres laboratorios de producción en los cuales se distribuye el total de personas.

Tabla 5. Recursos tecnológicos del proyecto ERP (Capital Humano)

Recursos	Cantidad
Computadora	15

4.4 Responsabilidades

La asignación de responsabilidades se hace teniendo en cuenta que el primer esfuerzo V&V lo realiza el desarrollador (analistas, diseñadores, programadores, en dependencia de la actividad que se esté realizando), que debe garantizar la calidad del proceso que ejecuta y del producto que se obtiene del mismo. El equipo de calidad del proyecto, en funciones de revisor, diseñador de pruebas, probador, está presente en todas las actividades del desarrollo y jugando su papel de contraparte obtiene un nivel de conocimientos vital para ejecutar de la manera más efectiva las pruebas y revisiones planificadas. La descripción detallada de las funciones precisas de cada rol con fines V&V se realiza dentro de cada tarea.

5. Procesos V&V

5.1 Proceso de Desarrollo

El proceso de desarrollo contiene las actividades y tareas del desarrollador. Se incluyen las actividades referentes al estudio preliminar, modelado de negocio, requisitos, diseño, implementación, pruebas, despliegue y soporte de los productos de software. Estas actividades están organizadas en Estudio preliminar V&V, Modelado de negocio V&V, Requisitos V&V, Diseño V&V, Implementación V&V, Prueba V&V, Despliegue V&V y Soporte V&V. Aquí se muestran las tareas más importantes asociadas a cada actividad.

5.1.1 Actividad: Estudio preliminar V&V

En la actividad de Estudio Preliminar se llevan a cabo las actividades relacionadas con la planeación del proyecto a un alto nivel, la evaluación de la factibilidad del proyecto y la legalización del mismo. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.

5.1.1.1 Tarea: Decidir la ejecución del proyecto

Descripción

Se realizan un conjunto de actividades tales como: presentar y discutir el informe técnico, comercial y económico así como decidir si se aprueba o se rechaza el proyecto.

En caso de que el proyecto se apruebe: se termina con la confección del proyecto técnico y se notifica al cliente. Encaso contrario se notifica al cliente

Roles y responsabilidades

Especialista de software: Decide si se aprueba o se rechaza el proyecto

Entradas requeridas

Informe del estado del arte, Informe técnico, económico y comercial

Salidas requeridas

Proyecto técnico, Registro del proyecto

5.1.2 Actividad: Modelado de negocio V&V

Durante la actividad de Modelado del negocio V&V se sientan las bases de todo el respaldo legal que debe tener el software, se identifican los procesos de negocio de la organización que se describen textualmente para una mejor comprensión de los mismos. Los esfuerzos V&V están encaminados fundamentalmente a verificar y validar que los procesos de negocio identificados reflejan correctamente el funcionamiento de la organización y que fueron documentados adecuadamente.

5.1.2.1 Tarea: Revisión de los procesos de negocios

Descripción

Se evalúa el modelado de negocio en cuanto a completitud, corrección y consistencia.

Completitud: Verificar y validar que los procesos identificados modelan la estructura y funcionamiento completo de la organización.

Corrección: Verificar que los procesos identificados están descritos correctamente y no se viola ningún paso dentro del flujo que describe la dinámica de los mismos.

Métodos y procedimientos

Los diagramas de BPMN, obtenidos a partir del entendimiento del negocio, que se realiza partiendo de los procesos identificados son sometidos a una prueba de validación con el proveedor de requisitos o funcional. De la misma manera la Descripción Textual del Negocio (DTN), se somete a una prueba de validación con el funcional, que valida y firma la propuesta quedando formalizado su total acuerdo con lo planteado en la misma.

Roles y responsabilidades

Analista principal: Identifica y describe muy brevemente los procesos de negocio.

Analista: Lleva a cabo la comprensión de todos los procesos de negocio y los describe en detalle, especificando el flujo de actividades de cada uno de ellos. De esto se obtiene la Descripción de proceso de negocio.

Proveedor de requisitos: Es el especialista funcional, el cual brinda información acerca del negocio así como también valida los diagramas de BPMN y la Descripción Textual del Negocio

Administrador de calidad: Realiza la revisión de la documentación, Pruebas Unitarias al sistema así como también participa en las pruebas de validación con el funcional

Entradas requeridas

Mapas de proceso del negocio, Descripción de procesos de negocio

Salidas requeridas

Reporte de tarea

5.1.3 Actividad: Requisitos V&V

En la actividad de Requerimientos V&V se definen los requerimientos funcionales y no funcionales del software, las interfaces, se construye la primera versión de la base de datos. El objetivo fundamental de los esfuerzos V&V es asegurar la corrección, completitud, exactitud, comprobabilidad y consistencia de los requerimientos.

5.1.3.1 Tarea: Evaluación de los requerimientos del software

Descripción

Evaluar los requerimientos del software en cuanto a corrección, consistencia, completitud, legibilidad y comprobabilidad.

Corrección: Verificar que todos los requerimientos plasmados en el Documento Visión estén incluidos en el Documento de Descripción de Requisitos. Verificar y validar que los requerimientos del software satisfacen las necesidades del usuario y que cumplen con los estándares, referencias, regulaciones, políticas, leyes gubernamentales y reglas del negocio.

Completitud: Verificar que el Documento de Descripción de Requisitos incluye todos los requerimientos correctamente descritos, teniendo en cuenta para cada funcionalidad

Legibilidad: Verificar que la documentación es legible, entendible y sin ambigüedades para la audiencia requerida. Verificar que la documentación ha sido escrita cumpliendo con el formato para la documentación definido para el proyecto. (Anexo E)

Comprobabilidad: Verificar y validar que existe un proceso concreto y finito que permita comprobar si se cumple el requisito.

Métodos y procedimientos:

Se aplica una revisión técnica para evaluar los criterios expuestos en la descripción de la tarea. Se realiza una prueba de validación con el funcional para garantizar la corrección de los requisitos. Siempre que surjan no conformidades se llevará a cabo un proceso de refinamiento de los requisitos para que estos vuelvan a ser sometidos a la revisión y a una prueba de validación.

Roles y responsabilidades

Analista: Identifica y describe las funcionalidades de sus módulos verificando constantemente su trabajo en cuanto a los criterios plasmados en la descripción de la tarea.

Proveedor de requisitos: Es el especialista funcional que valida los requerimientos descritos en el Documento de Descripción de Requisitos, verifica que satisfacen sus necesidades y que cumplen con los estándares, referencias, regulaciones, políticas, leyes gubernamentales y reglas del negocio

Administrador de calidad: Lleva a cabo la revisión técnica, entrega las no conformidades detectadas y repite la revisión a las funcionalidades refinadas.

Planificador: Planifica las actividades de verificación y validación de requisitos

Entradas requeridas

Documento Visión, Documento de Descripción de Requisitos, Modelo conceptual, Glosario de términos

Salidas requeridas

Reporte de tarea

5.1.4 Actividad: Diseño V&V

En la actividad de diseño los requerimientos del software son transformados en arquitectura y diseño detallado para cada componente del software. El diseño incluye bases de datos (entidades de dominio), interfaces de acceso a datos, interfaces de negocio y además la interfaz gráfica. El objetivo fundamental de la V&V es demostrar que el diseño es una correcta, precisa y completa transformación de los requerimientos del software y que no se han introducido errores.

5.1.4.1 Tarea: Evaluación del diseño del software

Descripción:

Se evalúa el diseño del software en cuanto a corrección y consistencia.

Corrección: Verificar y validar que el diseño del software se ajusta a las especificaciones de la arquitectura definida para el proyecto, además que cumpla con las convenciones de código establecidas, se ajuste a la estructura de paquetes y estén bien asignadas las responsabilidades de cada capa arquitectónica. Verificar que están definidas todas las funcionalidades necesarias para satisfacer cada caso de uso.

Consistencia: Verificar que las funcionalidades con objetivos similares que puedan ser parametrizadas en una sola no estén separadas. Verificar que están definidas las funcionalidades comunes para que sean implementadas solamente en un módulo.

Métodos y procedimientos

Se aplica un walk-through donde en una o varias sesiones de trabajo que tendrán lugar obligatoriamente con todos los arquitectos del proyecto donde cada arquitecto explica al resto la propuesta de solución que tiene para su módulo y en función de esto se toman todas las decisiones pertinentes para obtener un diseño correcto, completo y consistente. Se aplica una revisión técnica por parte del equipo de arquitectura de software que debe indicar los errores y explicar la forma idónea de proceder para solucionar cada uno.

Roles y responsabilidades

Arquitecto de sistema: Construye el diseño del software y participa en el walk-through y lleva a cabo la revisión e interviene en el diseño de la solución de ser necesario.

Entradas requeridas

Diagrama de componentes, Diseño de clases y Modelo de datos, Convenciones de código establecidas en el proyecto

Salidas requeridas

Reporte de tarea, Diseño correcto de la solución.

5.1.5 Actividad: Implementación V&V

5.1.5.1 Tarea: Evaluación del código y su documentación

Descripción

Se evaluará el código (y la documentación relativa a este) generado en cuanto a corrección, consistencia y completitud

Corrección: Se verifica que los componentes del código fuente cumplen con los estándares, regulaciones y políticas establecidas en el proyecto (convenciones de nombres). Verificar que se codifique respetando la estructura de paquetes global y para cada módulo.

Consistencia: Se verifica que exista la documentación adecuada para todo el código generado teniendo en cuenta las peculiaridades para cada componente:

Interfaces (Services, Manager), Recursos (XML y ficheros .json)

Completitud: Verificar que en el código fuente están presentes elementos como: funcionalidad (algoritmos, validaciones, manejo de excepciones)

Métodos y procedimientos

Se realiza un peer review por parte del Arquitecto de sistema junto al implementador del código que se revisa para evaluar los criterios de corrección, consistencia y completitud. Se aplica una inspección al código por parte del Arquitecto de sistema. Se aplica una revisión técnica para evaluar el criterio de legibilidad.

Roles y responsabilidades

Programador: Genera el código y participa en el peer review.

Arquitecto de sistema: Ejecuta la revisión de la documentación Supervisa el peer review. Ejecuta la inspección.

Entradas requeridas

Código fuente, Estándares de codificación (Anexo F), Convenciones de nombres definidos en el proyecto

Salidas requeridas

Reporte de tarea, Documento de No Conformidades

5.1.5. 3 Tarea: Ejecución y verificación de las pruebas unitarias

Descripción

Ejecutar las pruebas unitarias a nivel de acceso a datos y de negocio. Los resultados serán utilizados para validar que el software satisface los criterios de aceptación para la prueba.

Métodos y procedimientos

Se aplica una prueba unitaria a las unidades definidas y se evalúa el resultado contra los criterios de aceptación previamente definidos para esa prueba.

Roles y responsabilidades

Programador: Genera el código y realiza pruebas unitarias.

Administrador de calidad: Participa junto a los programadores en la aplicación de las pruebas. Evalúa la prueba atendiendo a los criterios de aceptación para la misma y elabora el Documento de No Conformidades

Entradas requeridas

Código fuente, Casos de prueba, Criterios de aceptación de la prueba

Salidas requeridas

Reporte de tarea, Documento de No Conformidades

5.1.6 Actividad: Prueba V&V

La actividad de pruebas V&V cubre las pruebas del software. El objetivo fundamental es asegurar que los requerimientos del software son satisfechos con la ejecución de pruebas de sistema, integración y aceptación; además de realizar una revisión completa a la documentación de usuario que se genera en el proyecto.

5.1.6.1 Tarea: Ejecución y verificación de las pruebas de aceptación

Descripción

Se realizan las pruebas de aceptación para comprobar que se satisfacen los requisitos pactados con el cliente. Se describen como pruebas funcionales sobre el total de requisitos distribuidos por procesos del negocio.

Métodos y procedimientos

Se ejecutan las pruebas de aceptación para comprobar el funcionamiento correcto del sistema bajo el criterio de aceptación del experto funcional.

Roles y responsabilidades

Especialista funcional .Ejecuta las pruebas y realiza las peticiones de cambios

Arquitecto: Asegura el correcto funcionamiento del equipamiento técnico del software, valora la significación de una petición de cambios, prepara a todo el equipo de prueba en el trabajo con el producto

Probador: Garantiza la correcta ejecución de las pruebas, asegura el entendimiento fluido entre la parte cliente y el equipo de desarrollo, recoge las no conformidades y solicitudes de cambios, y conforma el resumen de los resultados de las pruebas de aceptación

Analista: Apoya la correcta ejecución de las pruebas, valora la petición de un cambio, conforma el resultado de las pruebas de aceptación, y prepara al equipo de prueba en el trabajo con el producto.

Entradas requeridas

Versión estable del software, Casos de prueba

Salidas requeridas

Reporte de tarea, DNC, Documento de Petición de Cambios, Resultado de las pruebas de aceptación, Documento de Petición de Cambios, Resultado de las pruebas de aceptación

5.1.7 Actividad: Despliegue V&V

Esta actividad es la implantación del producto de software en el entorno previsto para su uso, así como la aceptación por parte del que lo adquiere. El objetivo fundamental es verificar y validar la corrección de la implantación del software en el ambiente correspondiente.

5.1.7.1 Tarea: Verificar la Instalación de los servidores

Descripción

CAPÍTULO II. PROPUESTA DEL PLAN DE V&V DE REQUISITOS

En el sistema integral de gestión Cedrux se utilizan dos servidores, uno para la aplicación y el otro para la base de datos, estos trabajaran sobre el sistema operativo Linux, específicamente en Ubuntu server 8.04. Se evalúa la configuración de instalación en cuanto a completitud y corrección.

Completitud: Verificar que el servidor de aplicación deberá tener instalados los programas y librerías: Apache 2.0, PHP 5, Php5-pgsql, Php-soap, Php5-gd, Php5-xsl, Php5-mcrypt. Verificar que el servidor de base de datos deberá tener instalado:Postgresql-8.3

Corrección: Verificar que la instalación se completa según los pasos definidos en el asistente de la instalación. Verificar una vez completada la instalación que está presente todos los componentes que debían instalarse.

Métodos y procedimientos

Se realiza una revisión técnica para evaluar la completitud de la configuración de instalación. Se ejecuta una prueba de instalación para evaluar el criterio de corrección detallado en la descripción de la tarea.

Roles y responsabilidades

Implantador: Prepara el paquete de instalación.

Entrenador: Revisa el paquete de instalación. Realiza la prueba de instalación.

Entradas requeridas

Paquete de instalación del software

Salidas requeridas

Reporte de tarea

5.1.7.2 Tarea: Verificar la Instalación técnica

Descripción

Evaluar el manual de instalación en cuanto a legibilidad y completitud.

Legibilidad: Verificar que el Manual de Instalación se ha escrito en un vocabulario correcto y claro, con la estructura y formato adecuados.

Compleitud: Verificar que contenga toda la información necesaria, para que el usuario siguiendo los pasos definidos para la instalación logre completarla con éxito.

Métodos y procedimientos

Revisión técnica para evaluar la legibilidad del manual. La prueba de instalación descrita en la tarea anterior debe ser realizada con el apoyo del Manual de Instalación, para evaluar la completitud del mismo.

Roles y responsabilidades

Especialista en implantación: Ejecuta la revisión del Manual

Especialista funcional: Ejecuta la prueba de instalación.

Entradas requeridas

Manual de Instalación

Salidas requeridas

Reporte de tarea

5.1.8 Actividad: Soporte V&V

Una vez iniciada la puesta en marcha del sistema viene una etapa de soporte que incluye, además de las tareas de asesoramiento técnico y reparaciones, las tareas de actualizar y adaptar los cambios que se realicen al software, es decir, incluye el desarrollo de las pruebas de regresión, desmonte de la línea base y archivo del producto así como el mantenimiento del producto.

5.1.8.1 Tarea: Análisis de la gestión de incidentes

Descripción

La gestión de incidentes tiene como objetivo resolver cualquier incidente que cause una interrupción en el servicio de la manera más rápida y eficaz posible, es decir, restablecer lo más rápidamente la calidad del servicio y no determinar cuáles han sido los orígenes y causas del mismo.

Métodos y procedimientos

Para supervisar el grado de satisfacción del cliente se hace uso de encuestas que permitan evaluar la percepción del cliente respecto a los servicios prestados.

Roles y responsabilidades

Operador: Registra y Cierra las incidencias en la herramienta de gestión. Emite un informe diario de aquellas incidencias que durante su estancia sean incorporadas o cambiadas de estado por el equipo de implantación o el equipo de desarrollo.

Responsable de la gestión de cambios: Gestiona los cambios solicitados por los clientes mediante peticiones de servicios en colaboración con la gestión de cambios y versiones.

Responsable de gestión de configuraciones: Colaborar con la gestión de configuración para asegurar la actualización de las BD correspondientes.

Responsable de gestionar el proceso de solución: Controla el proceso de solución de las incidencias en cada una de las líneas de desarrollo. Identifica, coordina y gestiona demoras en el proceso de solución de las incidencias, así como demoras en la respuesta de solución en los equipos de desarrollo

Entradas requeridas

Entrada de Incidente, Base de Conocimiento (KB), Base de Datos de Configuración (CMDB)

Salidas requeridas

Petición de Cambio (RFC). (Anexo I), Actualización de la información en la CMDB sobre los elementos de configuración (CI) implicados en el incidente, Restauración del servicio, Cierre del Incidente

5.1.8.2 Tarea: Análisis de la gestión de versiones

Descripción

La gestión de versiones es la encargada de la implementación y control de calidad de todo el software y hardware instalado en el entorno de producción. Debe colaborar estrechamente con la gestión de cambios y de configuraciones para asegurar que toda la información relativa a las nuevas versiones se integra adecuadamente en la CMDB de forma que ésta se halle correctamente.

Métodos y procedimientos

Se realizan pruebas del correcto funcionamiento de la versión en un entorno realista así como también pruebas de los procedimientos automáticos o manuales de instalación.

Roles y responsabilidades

Especialista funcional: Realizan las pruebas.

Entradas requeridas

Petición de Cambio (RFC), Base de Datos de Configuración (CMDB).

Salidas requeridas

Actualización de la Biblioteca de Software Definitivo (DSL), el Depósito de Hardware Definitivo (DHS) y la de la Base de Datos de Configuración (CMDB).

2.6 Conclusiones parciales

Se explicaron los aspectos que se tuvieron en cuenta para adaptar el estándar IEEE 1012-1998 a las circunstancias del esquema productivo del proyecto ERP, realizando previamente una caracterización general del mismo. Se desglosaron las partes que conforman el Plan V&V explicando cómo se aborda cada una dentro de la propuesta. Se explica la Actividad V&V y se exponen las características de todas las incluidas en el plan, además de esquematizar las tareas que las componen ilustrando los roles que intervienen en cada una, así como sus entradas y salidas requeridas. De la Tarea V&V se hace un desglose de los aspectos que definen su estructura revelando como se enfrentaron para confeccionar las tareas V&V ERP, haciendo énfasis en los métodos y procedimientos, por la variedad y especialización que tiene su aplicación dentro del plan. Luego de un análisis se puede concluir que la propuesta obtenida es capaz de introducir buenas prácticas y estandarizar los procesos de Verificación y Validación para el desarrollo del ERP.

CAPÍTULO 3. VALIDACIÓN DE LA PROPUESTA DEL PLAN DE VERIFICACIÓN Y VALIDACIÓN DE REQUISITOS

3.1 Introducción

Para validar la propuesta del Plan V&V de Requisitos que se definió en el capítulo anterior, se realizó un estudio de métodos y técnicas de validación como: la Matriz de Véster para identificar las causas, efectos y sus relaciones entre cada criterio emitido por los expertos, la Entrevista Cerrada a Expertos para obtener información precisa de los criterios emitidos por los expertos sobre la problemática planteada y por último el método de Criterios de Expertos (Técnica Delphi). Luego del análisis de estos métodos el proceso de validación se realizó mediante este último por la posibilidad de obtener información de forma independiente, de intercambio de información y de evitar evaluaciones superficiales.

3.2 El Método Delphi

El Método Delphi, cuyo nombre se inspira en el antiguo oráculo de Delphos, fue ideado originalmente a comienzos de los años 50 en el seno del centro de investigación estadounidense RAND Corporation¹¹ por Olaf Helmer y Theodore J. Gordon, como un instrumento para realizar predicciones sobre un caso de catástrofe nuclear. Desde entonces, ha sido utilizado frecuentemente como sistema para obtener información sobre el futuro. (ASTIGARRAGA, 2004)

Proporciona un medio para agregar sistemáticamente los juicios informados de un grupo de expertos dentro de su campo de conocimiento. Las opiniones de los participantes no se ven influenciadas por confrontaciones cara a cara. El anonimato de los panelistas evita presiones hacia la conformidad asociadas con reacciones ante las ideas de los otros y facilita la igualdad de los participantes evitando el efecto autoridad asociado a panelistas muy relevantes que podrían ser percibidos como referentes por el resto del grupo. (ARAGÓN, 2003)

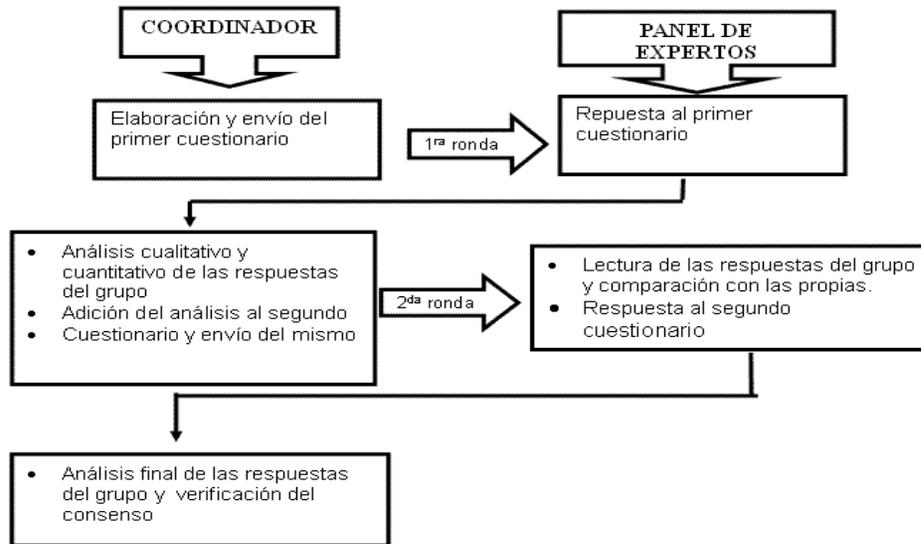


Figura 10. Diagrama del procedimiento del Método Delphi. Fuente: (TORRES et al. 2005)

De lo anteriormente dicho se puede concluir que el Método Delphi:

- ✓ Es un proceso iterativo ya que los expertos puedan ser consultados varias veces sobre la misma cuestión, es decir, garantiza que puedan volver a pensar sus respuestas, ayudados por la información que reciben de las opiniones del resto de los expertos.
- ✓ Mantiene el anonimato de los participantes, o, al menos de sus respuestas ya que éstas van directamente al grupo coordinador permitiendo desarrollar un proceso de grupo con expertos que no coinciden ni temporal ni espacialmente
- ✓ El intercambio de información entre los expertos no es libre, sino que se realiza a través de alguien que funciona como coordinador, con lo que se elimina toda información que no sea relevante.
- ✓ La respuesta es estadística del grupo, es decir, todas las opiniones forman parte de la respuesta final. Las preguntas están formuladas de forma que se pueda realizar un tratamiento cuantitativo y estadístico de las respuestas.

3.3 Aplicación del método

Según los investigadores María de Lourdes Bravo y José Joaquín Arrieta en su artículo *El Método Delphi. Su implementación en una estrategia didáctica para la enseñanza de las demostraciones geométricas*, hay 2 aspectos básicos a tener en cuenta para la aplicación del método:

1. La selección del grupo de expertos a encuestar: personas conocedoras, con reconocida competencia y con experiencia en el tema que garantice la confiabilidad de los resultados, creativos e interesados en participar.
2. Elaboración de los cuestionarios: tener en cuenta la teoría de la comunicación, con mecanismos que reduzcan los sesgos en las respuestas, preguntas claras, precisas e independientes. Suelen ser preguntas cuantitativas para calcular medias y rangos, y cualitativas para la justificación de sus opiniones. (BRAVO and ARRIETA, 2005)

3.3.1 Proceso de selección de los expertos

Se entiende por experto, tanto al individuo en sí como a un grupo de personas u organizaciones capaces de ofrecer valoraciones conclusivas de un problema en cuestión y hacer recomendaciones respecto a sus momentos fundamentales con un máximo de competencia. (URIZARRI and FERNÁNDEZ, 2005)

Se conformó un panel integrado por Especialistas del Centro de Calidad, Seguridad y Estándares (CES) del UCID, y otros especialistas del área productiva con experiencia en temas relacionados con la investigación, así como directivos con un alto nivel de conocimiento en temas de calidad de software.

Por la limitación en tiempo y recursos se decidió que la cantidad de expertos no fuera muy numerosa, aunque se corresponde con el intervalo sugerido por los autores José Ignacio Ruiz Olabuénaga y María Antonia Ispizua Uribarri en su libro *La descodificación de la vida cotidiana: métodos de investigación cualitativa*. (OLABUÉNAGA and URIBARRI, 1989)

Los expertos se seleccionaron según las características siguientes:

- ❖ Conocimientos acerca de los contenidos que sustentan la propuesta de solución:
 - Calidad de Software, Verificación y Validación, Requerimientos, Métodos de Verificación y Validación (estáticos y dinámicos), Proceso de Desarrollo del Software, Normas y estándares de calidad, Planificación
- ❖ Años de experiencia trabajando los temas que se señalaron anteriormente, así como también, prestigio en el colectivo de trabajo
- ❖ Capacidad de análisis y pensamiento lógico e integración a las actividades productivas

El número de expertos seleccionados fue siete con el título de Ingenieros en Ciencias Informáticas. Por la ocupación actual se desglosan en: un Asesor de Calidad del Centro de Desarrollo de Arquitecturas Empresariales (CDAE), un Jefe y dos Especialistas del Centro de Calidad, Seguridad y Estándares (CES) del UCID, un Analista de GEITEL, un Administrador de calidad del Centro de Informatización de la Gestión de Entidades (CEIGE) y un Especialista de calidad del Centro para la Excelencia en el Desarrollo de Proyectos Tecnológicos (Calisoft). (Anexo B)

Una vez confeccionado el panel se invitó a cada experto de manera personal para que participaran en el proceso de validación y aceptación de la propuesta. De esta manera, se termina el primer paso correspondiente a la selección del personal que integra el Panel de Expertos, por lo que se procede a la elaboración y aplicación del cuestionario.

3.3.2 Elaboración del cuestionario

Para el procesamiento y análisis de la información se tiene en cuenta el tipo de pregunta si es abierta o cerrada y se valora desde lo cualitativo o cuantitativo respectivamente. En el cuestionario elaborado se hizo énfasis en las preguntas de tipo cerradas, pero la mayoría de ellas incluye la posibilidad de que el experto emita su criterio sobre el tema tratado, algo característico de las preguntas abiertas. Esta forma de relacionar ambos tipos (abiertas y cerradas) permitió hacer en estos casos análisis cuantitativos y cualitativos sobre la misma pregunta. El siguiente cuadro detalla los objetivos que se persiguen con cada tipo de pregunta:

Tabla 6. Objetivos de los tipos de preguntas

Objetivo	Tipo de pregunta
Búsqueda de concordancia entre los expertos	Cerrada
Búsqueda de elementos comunes y esenciales en las valoraciones de los expertos	Abiertas

Para la elaboración de las preguntas se tuvieron en cuenta cinco objetivos generales:

1. Demostrar la importancia de la detección temprana de errores y su implicación en la garantía de la calidad de un software a través de la planificación V&V, combinando la aplicación de mecanismos de control estáticos y dinámicos. Preguntas 1, 2, 3 y 7
2. Demostrar que las actividades propuestas son necesarias y suficientes para darle cumplimiento a los objetivos de los procesos de Verificación y Validación en el proyecto ERP. Pregunta 6
3. Demostrar que el plan es el resultado de la correcta adecuación del estándar IEEE 1012-1998 a las características del ERP. Preguntas 4, 5, 8
4. Demostrar que se hace una distribución correcta de roles, responsabilidades, métodos y procedimientos para darle cumplimiento a las tareas del plan. Preguntas 9 y 10
5. Demostrar que la propuesta constituye un aporte a la mejora y estandarización de los procesos de Verificación y Validación en el proyecto ERP. (Este objetivo se hace explícito en la pregunta 11, pero se encuentra implícito en todas las demás)

Los expertos que conforman el panel recibieron un resumen de la propuesta de solución como documentación primaria para responder los temas encuestados, además del cuestionario con un total de once preguntas con las características explicadas anteriormente. Se realizó una sola ronda de preguntas y luego se prosiguió a analizar los resultados.

3.4 Resultados del proceso de aplicación del método

La tabulación de los resultados se efectuó por cada uno de los objetivos descritos anteriormente, sobre la base de las evaluaciones positivas y negativas emitidas por los expertos. Se tabularon en gráficos que muestran la concordancia entre los expertos, donde por el eje “x” se tiene la cantidad de expertos que dieron una u otra respuesta (positiva o negativa) y por el eje “y” los objetivos específicos relacionados. Para el cumplimiento de estos últimos, la respuesta a la pregunta asociada debe ser evaluada positivamente por más del 75% de los expertos. A continuación se detallan los resultados:

Objetivo general # 1: Demostrar la importancia de la detección temprana de errores y su implicación en la garantía de la calidad de un software a través de la planificación V&V combinando la aplicación de mecanismos de control estáticos y dinámicos. Se desglosa en cuatro objetivos específicos asociados a las

preguntas 1, 2, 3 y 7, para las cuales las evaluaciones positivas y negativas se toman de la forma en que se muestra en la Tabla 7.

Tabla 7. Evaluaciones positivas y negativas del Objetivo general # 1

Pregunta	Objetivo específico relacionado	Evaluación positiva	Evaluación negativa
1	Corroborar la importancia de la detección temprana de errores durante el desarrollo de un software	Ponderación en valores de 4 y 5	Ponderación en valores de 1 a 3
2	Ratificar la importancia de aplicar revisiones en todas sus variantes para garantizar la calidad de un software	Ponderación en valores de 4 y 5	Ponderación en valores de 1 a 3
3	Demostrar la importancia de planificar V&V dentro de un proyecto de desarrollo de software	Respuesta de Si	Respuesta de No
7	Reafirmar la insuficiencia de solo aplicar pruebas en funciones V&V.	Respuesta de No	Respuesta de Si

El resultado luego de la tabulación de las respuestas de cada experto y la evaluación según la tabla anterior se muestra en el siguiente gráfico:

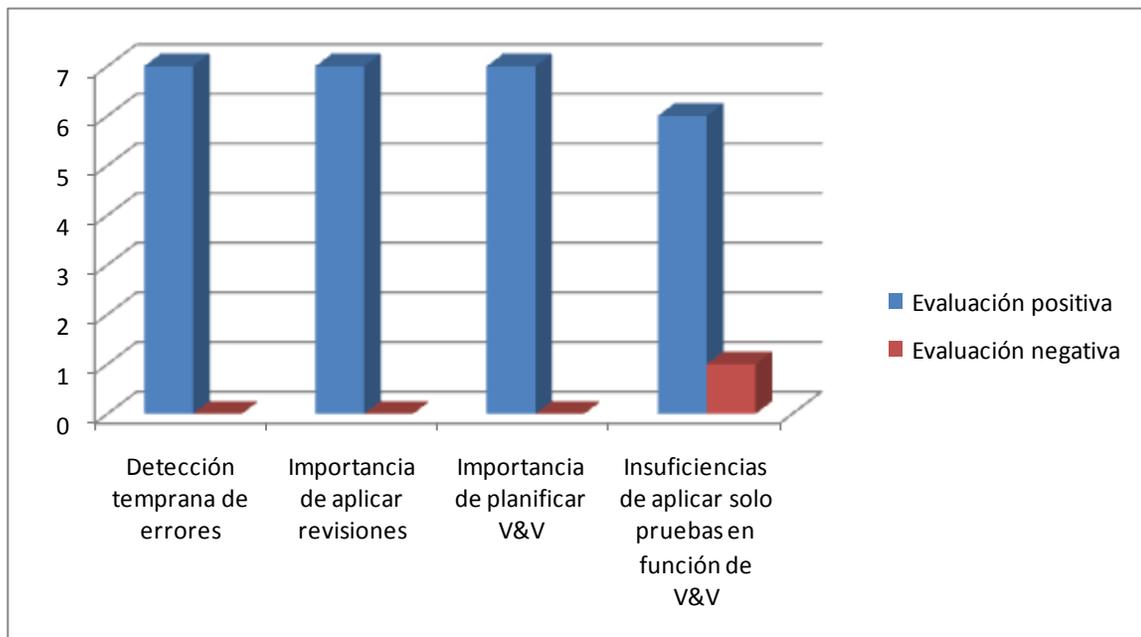


Figura 11. Evaluaciones de los expertos: Objetivo general #1

Objetivo general # 2: Demostrar que las actividades propuestas son necesarias y suficientes para darle cumplimiento a los objetivos de los procesos de Verificación y Validación en el proyecto ERP. Se desglosa en dos objetivos específicos asociados a la pregunta 6 para la cual las evaluaciones positivas y negativas se toman de la forma en que se muestra en la Tabla 8.

Tabla 8. Evaluaciones positivas y negativas del Objetivo general # 2

Pregunta	Objetivo específico relacionado	Evaluación positiva	Evaluación negativa
6	Demostrar que las actividades propuestas son necesarias para cumplir los objetivos V&V en el ERP	Todas las actividades están ponderadas de 4 y 5	Al menos una actividad está ponderada de 1 a 3
6-a	Demostrar que las actividades propuestas son necesarias para cumplir los objetivos V&V en el ERP	Respuesta de No	Respuesta de Sí

El resultado luego de la tabulación de las respuestas de cada experto y la evaluación según la tabla anterior se muestra en el siguiente gráfico:

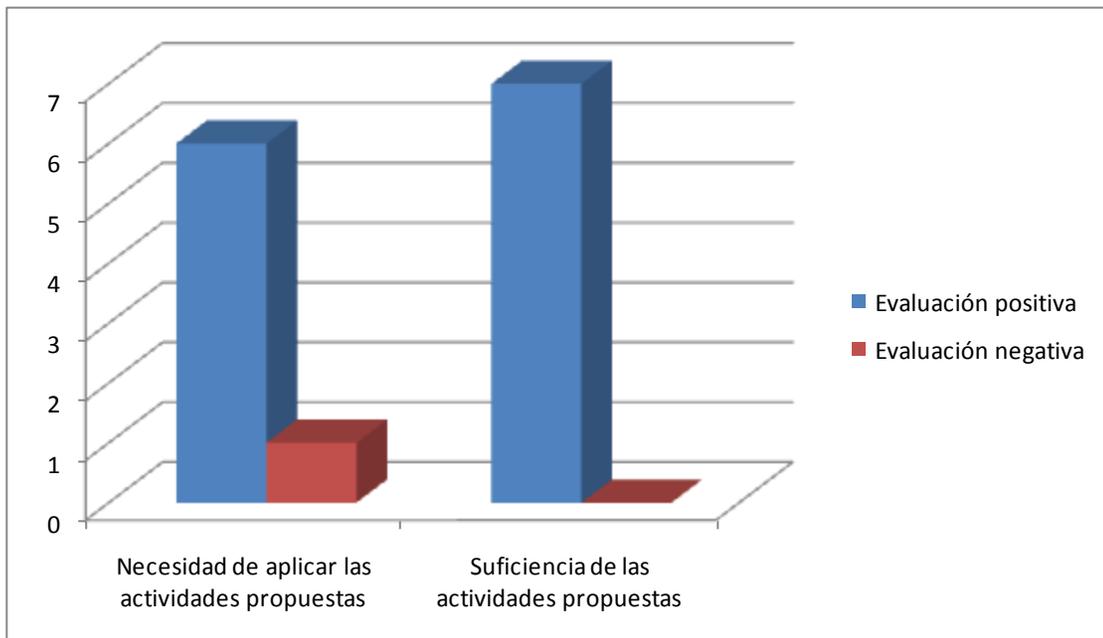


Figura 12. Evaluaciones de los expertos: Objetivo general #2

Objetivo general #3: Demostrar que el plan es el resultado de la correcta adecuación del estándar IEEE 1012-1998 a las características del ERP. Se desglosa en tres objetivos específicos asociados a las preguntas 4,5 y 8, para las cuales las evaluaciones positivas y negativas se toman de la forma en que se muestra en la Tabla 9.

Tabla 9. Evaluaciones positivas y negativas del Objetivo general # 3

Pregunta	Objetivo específico relacionado	Evaluación positiva	Evaluación negativa
4	Demostrar que la guía de adopción de la norma es correcta	Respuesta Adecuada	Respuesta No Adecuada
5	Demostrar que las actividades propuestas son necesarias para cumplir los objetivos V&V en el ERP	Respuesta de No	Respuesta de Sí
8	Demostrar que el cronograma tipo propuesto expresa claramente los aspectos necesarios de las actividades y tareas para la aplicación del plan en el ERP	Selección correcta de más de 2 aspectos	Selección de 2 o menos aspectos

El resultado luego de la tabulación de las respuestas de cada experto y la evaluación según la tabla anterior se muestra en el siguiente gráfico:

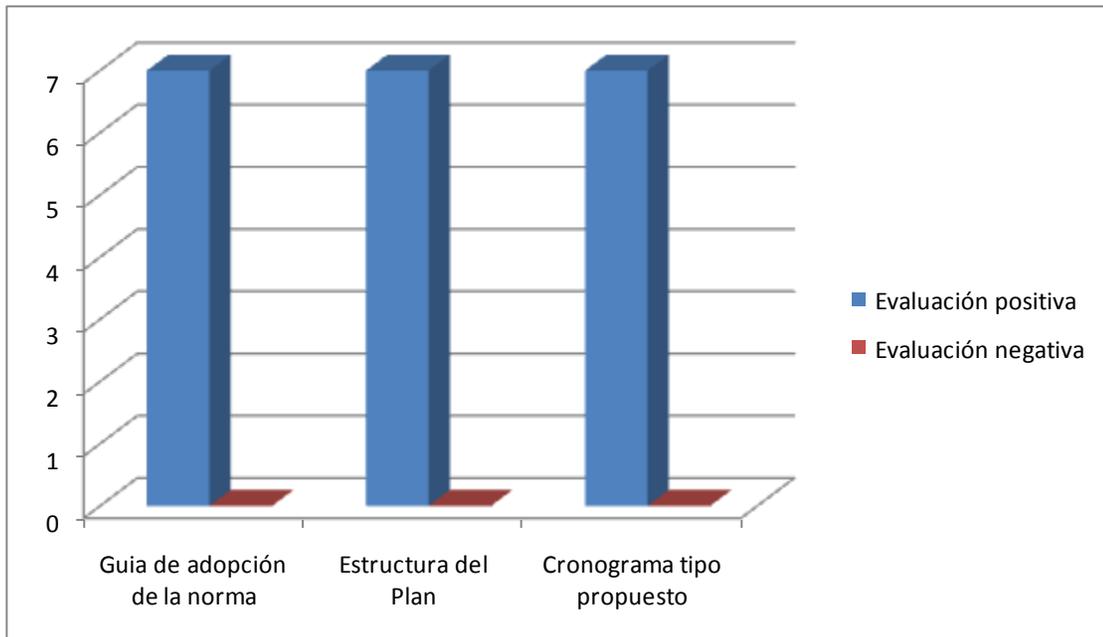


Figura 13. Evaluaciones de los expertos: Objetivo general #3

Objetivo general # 4: Demostrar que se hace una distribución correcta de roles, responsabilidades, métodos y procedimientos para darle cumplimiento a las tareas del plan. Se desglosa en dos objetivos específicos asociados a las pregunta 9 y 10, para las cuales las evaluaciones positivas y negativas se toman de la forma en que se muestra en la Tabla 10.

Tabla 10. Evaluaciones positivas y negativas del Objetivo general # 4

Pregunta	Objetivo específico relacionado	Evaluación positiva	Evaluación negativa
9 y 10-b	Demostrar que la asignación de roles y responsabilidades es correcta para las tareas del plan.	Respuestas de Correcta y Aceptable	Respuesta de Incorrecta
10	Demostrar que la aplicación de métodos y procedimientos para darle cumplimiento a las tareas es correcta.	Respuestas de Correcta y Aceptable	Respuesta de Incorrecta

El resultado luego de la tabulación de las respuestas de cada experto y la evaluación según la tabla anterior se muestra en el siguiente gráfico:

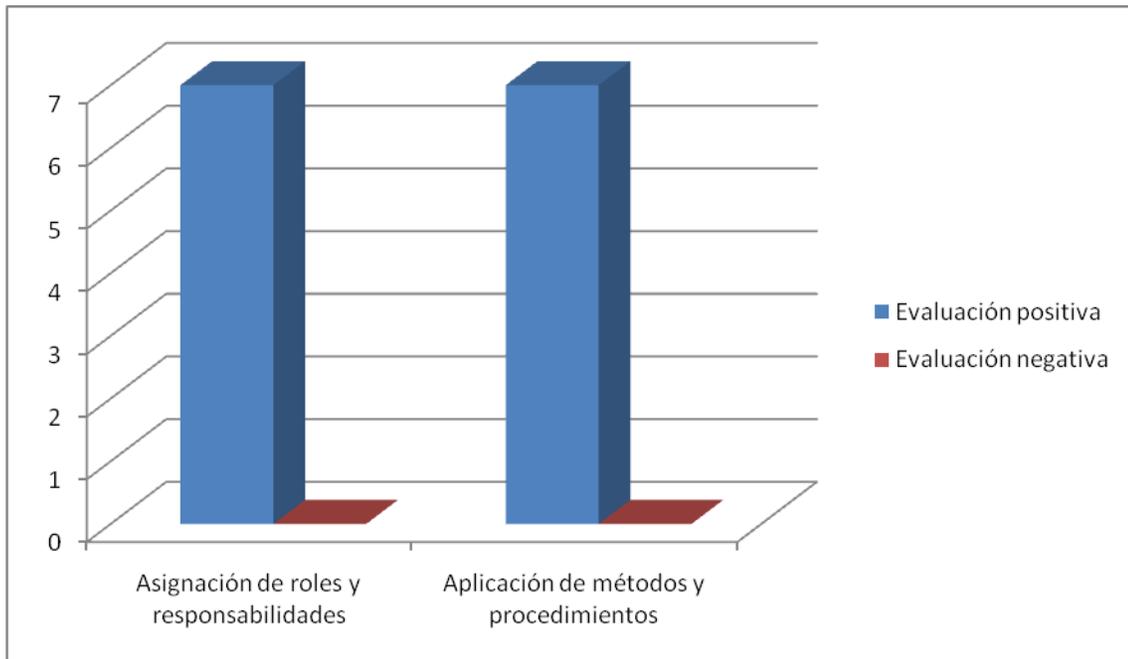


Figura 14. Evaluaciones de los expertos: Objetivo general #4

Objetivo general # 5: Demostrar que la propuesta constituye un aporte a la mejora y estandarización de los procesos de Verificación y Validación en el proyecto ERP. Este objetivo como se explica anteriormente se hace explícito en la pregunta 11, pero tiene una dependencia parcial del cumplimiento de los cuatro objetivos generales que le preceden, la manera en que se toman las evaluaciones positivas y negativas se muestra en la Tabla 11.

Tabla 11. Evaluaciones positivas y negativas del Objetivo general # 5

Pregunta	Objetivo específico relacionado	Evaluación positiva	Evaluación negativa
11	Demostrar que la propuesta constituye un aporte a la mejora y estandarización de los procesos de Verificación y Validación en el proyecto ERP	Respuesta de Si	Respuesta de No

El resultado luego de la tabulación de las respuestas de cada experto y la evaluación según la tabla anterior se muestra en el siguiente gráfico:

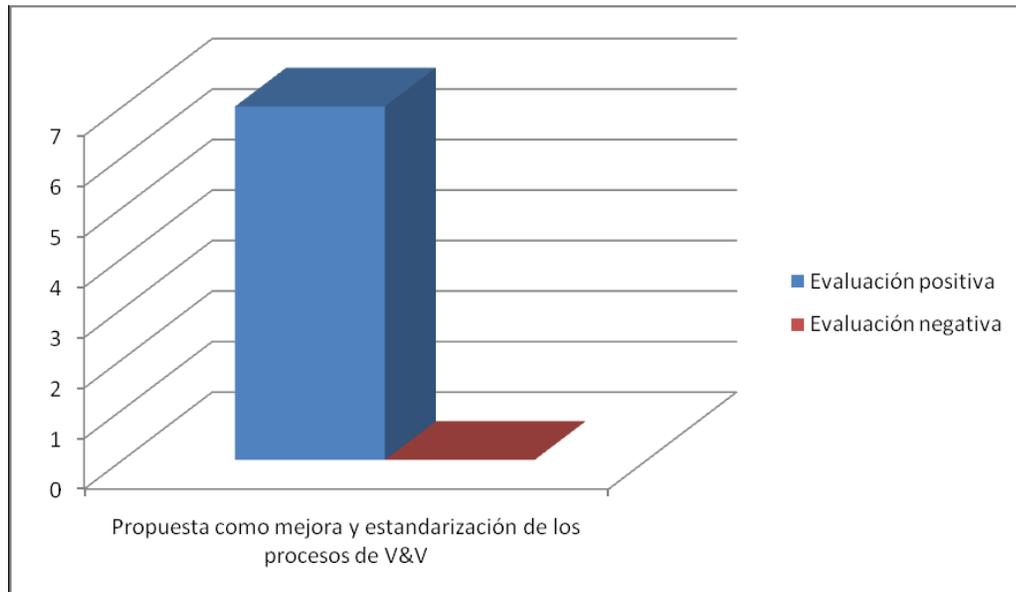


Figura 15. Evaluaciones de los expertos: Objetivo general #5

Sobre los diferentes temas tratados en el cuestionario los expertos emitieron opiniones como las que se señalan a continuación:

- ✓ La detección temprana de errores ahorra tiempo de corrección en otras etapas del desarrollo y permite que queden correctamente especificados los requisitos de acuerdo con las exigencias del cliente. Esto ayuda a que el producto termine con mejor calidad y se garantiza menor tiempo y esfuerzo en la fase de mantenimiento.
- ✓ Todas las actividades que se realizan en el desarrollo de un proyecto tienen que estar planificadas. Los procesos de Verificación y Validación han de estar planificados para que se realicen con la mejor calidad posible y no se afecte el cronograma del proyecto.
- ✓ Con una planificación objetiva y bien detallada se está asegurando un buen término y entrega en tiempo del producto a los clientes, y con la calidad requerida por lo que la planificación de los procesos de Verificación y Validación garantiza obtener un producto final con calidad y con la cantidad mínima de no conformidades.
- ✓ La adecuación de la norma IEEE 1012-1998 al proyecto es muy importante, ya que la IEEE es una organización que tiene bien definidos todos sus estándares y con gran reconocimiento a nivel

internacional y esto crea confianza en la propuesta por lo que la guía de adopción de la norma asegura la confección de un buen Plan V&V mediante la adecuación al proyecto del estándar IEEE 1012-1998.

- ✓ La estructura tiene un orden lógico y una secuencia que permite el entendimiento del plan, donde están desglosadas cada una de las actividades con sus tareas muy bien descritas y son especificados claramente los métodos que se utilizarán, las responsabilidades de los roles, así como las entradas que se requieren y las salidas esperadas.
- ✓ Resulta muy importante el cronograma tipo propuesto para conocer en qué momento del desarrollo se deben ejecutar las tareas, así como el orden de ejecución de las mismas.
- ✓ Si el plan se aplica al 100 % en el proyecto, el resultado debe ser satisfactorio y se recomienda que se extienda no solo a los proyectos de la facultad donde se desarrolla, sino a toda la universidad.
- ✓ Si se aplica el plan acorde a lo definido en esta estrategia, debe cumplir su objetivo que es estandarizar los procesos de Verificación y Validación del proyecto productivo.
- ✓ A través de este plan el proyecto garantiza la calidad de su desarrollo y evita que el sistema y la documentación asociada lleguen con errores a las pruebas de liberación y aceptación.

3.5 Conclusiones parciales

La aplicación del cuestionario a los siete expertos seleccionados arrojó resultados satisfactorios:

- Todas las preguntas fueron respondidas positivamente por más del 75% de los expertos.
- Los objetivos específicos asociados a cada pregunta fueron cumplidos.
- Los cinco objetivos generales a partir de los cuales se elaboraron las preguntas fueron cumplidos satisfactoriamente.

Se puede concluir que la propuesta fue validada por la totalidad de los miembros del panel, algo que corroboran las opiniones emitidas.

CONCLUSIONES

Como conclusiones generales de la investigación se muestran alcanzados los objetivos propuestos satisfactoriamente:

- *Incorporar nuevas actividades de Verificación y Validación de Requisitos a las realizadas en el proceso de desarrollo del ERP - Cuba, según las normas internacionales existentes.* Para este objetivo se realizó la tarea de identificar a partir de un estudio del proceso de desarrollo del proyecto ERP, qué actividades de Verificación y Validación se realizan y cómo se ejecutan. Otra tarea realizada para satisfacer dicho objetivo fue identificar los principales estándares y modelos que sirvieron de soporte a la propuesta de solución. Luego del estudio realizado y mostrado en el Capítulo I se definió basar la propuesta en el estándar 1012-1998, lo cual permitió identificar las nuevas actividades y tareas a incorporar según las características del proyecto.
- *Conformar la propuesta de un Plan de V&V que incluya buenas prácticas de Verificación y Validación de Requisitos ajustado a las características del proyecto.* Para darle cumplimiento a este objetivo, en el Capítulo I del trabajo investigativo se estudiaron a profundidad los procesos de Verificación y Validación y las principales definiciones dadas por diferentes autores, además de los métodos V&V que se aplican a lo largo del ciclo de vida de un software que se puedan ajustar al entorno del proyecto. Sobre la base de estos conocimientos y con el estudio del estándar seleccionado, en el Capítulo II se explicó la confección de la propuesta de Plan V&V.
- *Validar la propuesta de solución.* Se realizó utilizando el Método Delphi donde a través de la tabulación de los resultados obtenidos de la aplicación del cuestionario al Panel de Expertos, proceso que se describe en el Capítulo III, se puede llegar a la conclusión de que fue cumplido satisfactoriamente el objetivo, por lo que se cuenta con un documento válido que estandariza y guía los procesos de Verificación y Validación en el proyecto.

Por lo anteriormente descrito se concluye el cumplimiento del objetivo general de la investigación de elaborar un Plan de (V&V) que facilite la introducción de buenas prácticas en los procesos de Verificación y Validación de Requisitos en el proyecto ERP- Cuba.

RECOMENDACIONES

No se puede olvidar que el objetivo fundamental de todo proyecto de software es entregar un producto de calidad que se ajuste a los requerimientos del cliente y que este quede satisfecho con la solución recibida. Tomando en cuenta esto y con la importancia que revisten los procesos de Verificación y Validación para el desarrollo de software, se recomienda lo siguiente:

- ✓ Completar el Plan para las etapas de Operación y Mantenimiento del Software.
- ✓ Exigir el Plan V&V como documento obligatorio para todo proyecto productivo de la universidad.
- ✓ Realizar los ajustes necesarios al plan de manera que pueda ser aplicado en el resto de los proyectos de la universidad.
- ✓ Capacitar a los equipos de calidad de los proyectos de la universidad sobre los temas de Verificación y Validación de Requisitos.

REFERENCIAS BIBLIOGRÁFICAS

1. (ABRAN and MOORE 2004) ABRAN, A. and J. W. MOORE. *Guide to the Software Engineering Body of Knowledge, 2004.*
2. (ANTONIO 1999) ANTONIO, A. D. *Gestión, control y garantía de la calidad del software, 1999.*
3. (ARAGÓN 2003) ARAGÓN, S. *Método Delphi, 2003.* 2.
4. (ASTIGARRAGA 2004) ASTIGARRAGA, E. *El Método Delphi, 2004.* 14.
5. (BERMUDEZ and CORONA 2002) BERMUDEZ, G. S. and M. J. CORONA. *Diseño de un estándar de verificación y validación para asegurar la calidad del software, 2002.*
6. (BOOCH et al. 2000) BOOCH, G.; I. JACOBSON, et al. *El Proceso Unificado de desarrollo de software.* 2000. 458 p. 84-7829-036-2
7. (BRAVO and ARRIETA 2005) BRAVO, M. D. L. and J. J. ARRIETA. *El Método Delphi. Su implementación en una estrategia didáctica para la enseñanza de las demostraciones geométricas.: Revista Iberoamericana de Educación, 2005.* 11.
8. Figueroa, María Antonieta Abud. *Calidad en la Industria del Software. La Norma ISO-9126.*
9. (FOUNDATION 2008) FOUNDATION, W. *Pruebas de Software, 2008.* [2008]. Disponible en: http://es.wikipedia.org/wiki/Pruebas_de_software
10. (GUZMÁN et al. 2006) GUZMÁN, J. G.; A. D. A. SECO, et al. *TOP 10 de factores que obstaculizan la mejora de los procesos de verificación y validación en organizaciones intensivas en software, 2006.*
11. (IEEE 1997) ---. *Standard for Software Reviews. Std 1028, 1997.*
12. (IEEE 1998) ---. *Standard for Software Verification and Validation. Std 1012, 1998.*
13. (IEEE 1990) ---. *Standard Glossary of Software Engineering Terminology, 1990.*
14. (IEEE/EIA 1997) IEEE/EIA. *Industry Implementation of International Standard ISO/IEC 12207: 1995, 1997.*
15. (INSTITUTE 2006) INSTITUTE, S. E. *CMMI for Development, Version 1.2. 2006.* 573 p.

16. ISW. 2008. Clases de Ingeniería de software I y II. 2010.
17. Javo. 2010. QA & Testing en SCRUM. Argentina – Software y Servicios Informáticos. [En línea] 24 de Enero de 2010. [Citado el: 08 de Febrero de 2010.] <http://cbasga.wordpress.com/tag/aseguramiento-de-calidad/>.
18. Jean-Philippe Lang. 2006. Redmine. [En línea] 2010. <http://www.redmine.org/>.
19. (LANDETA et al. 2003) LANDETA, J.; J. M. D. ANTONIO, et al. *Aplicación del Método Delphi en la elaboración de la tabla simétrica de las tablas input-output 2001 de Catalunya*, 2003. 28.
20. (LEBRÚN et al. 2008) LEBRÚN, C. V.; L. R. PRIETO, et al. *Mejores prácticas para el establecimiento y aseguramiento de la calidad del software*, 2008.
21. (LEON 2006) LEON, F. *Ingeniería del Software*, 2006. 22.
22. Lovelle, Juan Manuel Cueva. 1999. *Calidad del Software*. 1999.
23. Malevski, Yoram. 1995. *Manual de Gestión de la Calidad Total a la Medida*. 1995.
24. (MAÑAS 1994) MAÑAS, J. *Prueba de Programas*, 1994.
25. (MARTÍNEZ 2004) MARTÍNEZ, C. A. *La importancia de las Herramientas de V&V*, 2004.
26. (OLABUÉNAGA and URIBARRI 1989) OLABUÉNAGA, J. I. R. and M. A. I. URIBARRI. *La descodificación de la vida cotidiana: métodos de investigación cualitativa*. 1989. 241 p. 84-7485-126-2
27. (PMBOK 2005) PMBOK. *Guía de los Fundamentos de la Dirección de Proyectos*, 2005.
28. (PRESSMAN 2002) PRESSMAN, R. S. *Ingeniería de Software. Un enfoque práctico*. 5ta. 2002. 614 p.
29. (QUISPE-OTAZU 2007) QUISPE-OTAZU, R. *Proceso de Desarrollo de Software*, 2007. [2008]. Disponible en: <http://www.rodolfoquispe.org/blog/proceso-de-desarrollo-de-software.php>
30. Santana, Ildian Guzmán. 2010. *Plan de Aseguramiento de la Calidad para proyectos de software sobre la Arquitectura MDA (Model Driven Architecture)*. 2010

31. (TIAN 2005) TIAN, J. *Software Quality Engineering. Testing, Quality Assurance, and Quantifiable Improvement*, 2005.
32. (TORRES et al. 2005) TORRES, M.; C. DANDICOURT, et al. *Funciones de enfermería en la atención primaria de salud. Revista Cubana de Medicina General Integral*, 2005.
33. (URIZARRI and FERNÁNDEZ 2005) URIZARRI, L. A. R. and A. M. T. FERNÁNDEZ. *Algunas consideraciones acerca del método de evaluación utilizando el criterio de expertos*, 2005.
34. (WEITZENFELD 2006) WEITZENFELD, A. *Ingeniería de software orientada a objetos con Java e Internet*, 2006.

GLOSARIO DE TÉRMINOS

Aseguramiento de la calidad del software: El aseguramiento de la calidad es la aplicación de actividades planificadas y sistemáticas relativas a la calidad, para asegurar que el proyecto emplee todos los procesos necesarios para cumplir con los requisitos.

Auditoría: Es el examen independiente de un producto de trabajo o grupo de productos de trabajo para evaluar el cumplimiento con especificaciones, estándares, acuerdos contractuales u otro criterio.

Calidad: La capacidad de un grupo de características inherentes a un producto, componente o proceso de satisfacer los requerimientos del cliente.

Caso de prueba: Un conjunto de entradas de prueba, condiciones de ejecución, y los resultados esperados desarrollados para un objetivo particular.

Ciclo de vida del software: Es el período de tiempo que comienza cuando un producto de software es concebido y termina cuando ya no está disponible para su uso.

CMMI: Modelo Integrado de Madurez de la Capacidad.

Comprobación de escritorio (desk checking): Consiste en examinar a mano e individualmente el objeto que se acaba de desarrollar. Es el método tradicional para analizar un programa. Se debe aplicar a los requisitos, especificaciones de diseño y código según se van desarrollando.

Criterios de aceptación: Los criterios que un producto o componente del producto debe satisfacer a ser aceptado por un usuario, cliente, u otra entidad autorizada.

Criticidad: El grado de impacto que un requisito, módulo, error, fallo, u otro elemento tiene en el desarrollo o funcionamiento del sistema.

Diseño de Pruebas: Documentación que especifica los detalles del método de prueba para una función de software o una combinación de funciones de software y la identificación de las correspondientes pruebas.

ERP: Planificación de Recursos Empresariales.

Error: Acción humana que produce un resultado incorrecto. Por ejemplo, una acción incorrecta por parte de un programador u operador.

Falla: La incapacidad de un sistema o componente para desempeñar sus funciones requeridas dentro de los requerimientos de rendimiento específicos.

IEEE: Instituto de Ingenieros Eléctricos y Electrónicos.

IEEE 1012-1998: Estándar para la Verificación y Validación de Software.

IEEE 1028-1997: Estándar para Revisiones al Software.

IEEE 610.12: Glosario de Terminologías de Ingeniería de Software.

Inspección: Es un examen visual de un producto de software para detectar e identificar anomalías, incluyendo errores y desviaciones de los estándares y especificaciones. Las inspecciones son exámenes por pares dirigido por facilitadores imparciales que están capacitados en técnicas de inspección.

ISO/IEC 12207: Procesos del Ciclo de Vida del Software.

IV&V: Verificación y Validación Independiente.

Plan V&V: Es un plan que guía el comportamiento de la verificación y la validación en un software.

PMBOK: Fundamentos de la Gestión de Proyectos.

Práctica específica (CMMI): Un modelo de componentes que es considerado importante en el logro de la meta específica asociada. La práctica específica describe las actividades previstas en el logro de los objetivos específicos de un área de proceso.

Procedimientos de Prueba: Las instrucciones detalladas para la puesta en marcha, ejecución y evaluación de resultados para un determinado caso de prueba.

Proceso: Un proceso (del latín processuss) es un conjunto de actividades o eventos que se realizan o suceden con un determinado fin.

Proceso de Desarrollo: Es la definición del conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto, a modo de plantilla que explica los pasos necesarios para terminar el proyecto.

Prueba de software: El proceso de analizar un software para detectar las diferencias entre las condiciones existentes y las requeridas además evaluar las características del sistema.

Requisito: Es una descripción de una condición o capacidad que debe cumplir un sistema.

Revisión: Es un proceso o una reunión durante la cual un producto de software es presentado al personal de proyectos, gerentes, usuarios, clientes, representantes de los usuarios, u otras partes interesadas para que formulen sus observaciones o lo aprueben.

Revisión por pares o iguales (peer review): Es la evaluación científica en la búsqueda o propósitos para la competencia y originalidad por expertos calificados que buscan crear productos de la misma forma.

Revisión técnica: Es una evaluación sistemática de un producto de software por un equipo de personal calificado, que examina la idoneidad del producto de software para su uso e identifica discrepancias con las especificaciones y estándares. Las revisiones técnicas también pueden proporcionar recomendaciones de alternativas y el examen de las distintas alternativas.

SEI: Instituto de Ingeniería de Software.

Validación: El proceso de evaluar un sistema o componente durante o al final del proceso de desarrollo para determinar si este satisface los requerimientos especificados. **Verificación:** El proceso de evaluar un sistema o componente para determinar si los productos de una fase satisfacen las condiciones impuestas al inicio de la misma.

Verificación y Validación (V&V): El proceso de determinar si los requisitos para un sistema o componente son completos y correctos, los productos de cada fase de desarrollo cumplen con los requisitos o condiciones impuestas por la fase anterior, y el sistema o componente final cumple con los requisitos especificados.

Walk-through: Es una técnica de análisis estático en el cual el diseñador o programador guía a miembros del equipo de desarrollo y otras partes interesadas a través del producto de software, los participantes realizan preguntas y comentarios acerca de posibles errores, violación de los estándares de desarrollo y otros problemas.