

GERESOFT

Gestión y verificación de requisitos

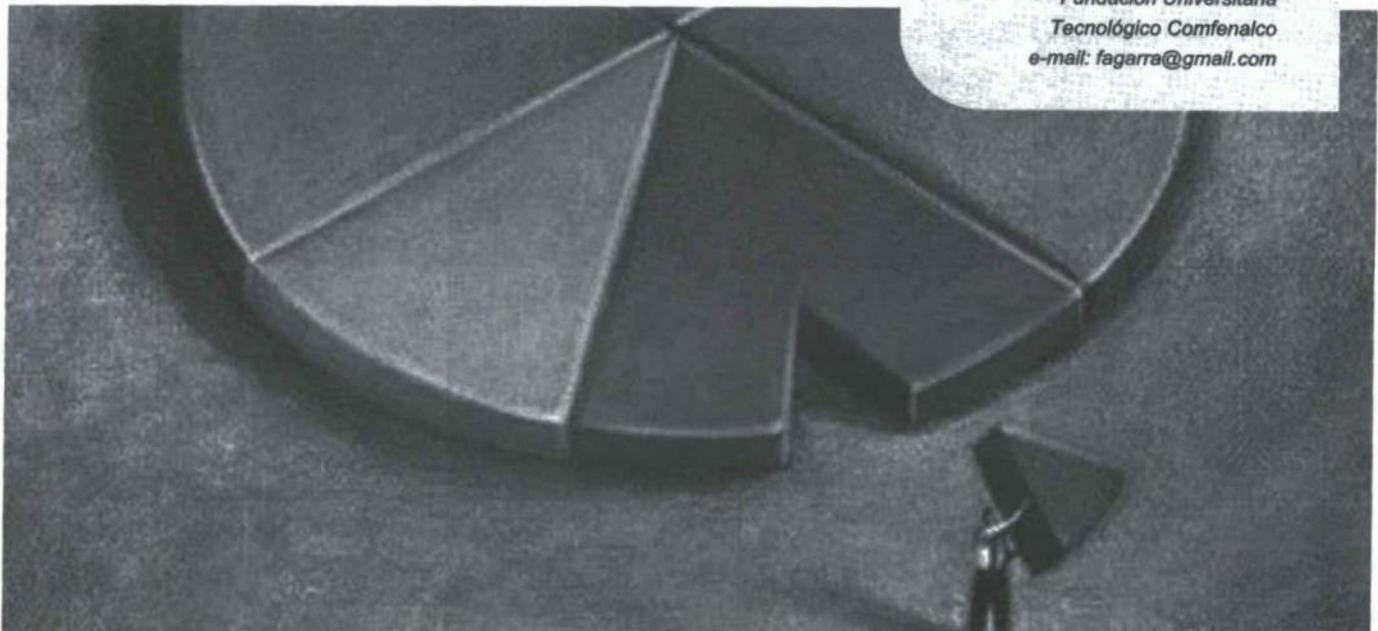
Geresoft
Management and verification of requirements

PLINIO PUELLO

*Magister (c) en Software Libre.
Universidad San Buenaventura
e-mail: pliniopuello@gmail.com*

FABIO GARCÍA

*Magister (c) en Software Libre.
Fundación Universitaria
Tecnológico Comfenalco
e-mail: fagarra@gmail.com*



RESUMEN: En este artículo se presenta la herramienta GERESOFT, como un Software para la Verificación y Validación de la Gestión de Requisitos, tomando como base para su diseño y construcción lineamientos del Software Libre, estándares de la IEEE y conceptos relacionados con la V&V (Verificación y Validación). Para realizar el proceso de V&V se utilizaron las Revisiones como Técnica de V&V bajo la orientación del Análisis Estático, y desde el punto de vista de la Ingeniería de Software como insumo para aplicar el conocimiento generado con este desarrollo se utilizó la Ingeniería de Requisitos, la cual es el proceso fundamental que permite planificar, administrar y desarrollar un proyecto de software exitoso desde su fase inicial.

PALABRAS CLAVE: Verificación y Validación, Ingeniería de Requisitos, Análisis Estático, Revisiones.

ABSTRACT: In this article we introduce GERESOFT tool as software for verification and validation of management requirements, whose design and construction is based on guidelines of Free Software, IEEE standards and concepts related to the V & V (Verification and Validation). To make the process of V & V, technical reviews of V & V under the guidance of Analysis Static were used, and from the viewpoint of software engineering as input to apply the knowledge generated by this development, requirements Engineering was used, which is the fundamental process that lets you plan, manage and develop a successful software project from its initial phase.

KEYWORDS: Verification and Validation, Requirements Engineering, Static Analysis, Reviews.

1. INTRODUCCIÓN

Una reflexión lamentable sobre la industria del software es que muchos usuarios tienen pocas expectativas sobre su software y no se sorprenden cuando éste falla durante su uso. Están dispuestos a aceptar estos fallos del sistema cuando los beneficios de su uso son mayores que sus desventajas. Sin embargo, la tolerancia de los usuarios a los fallos de los sistemas está decreciendo desde los años 90[1]. La anterior situación ha obligado a la industria del software y a los desarrolladores a mejorar los esfuerzos y recursos destinados al aseguramiento de la calidad de los productos durante el proceso de construcción de los mismos.

Dentro de estos esfuerzos, se destaca los procesos de Verificación y Validación (V&V), los cuales permiten evaluar si la aplicación de software fue desarrollada correctamente y cumple con los requerimientos del usuario.

1.1 Verificación y Validación

La Verificación en la Ingeniería de Software es entendida como comprobar que una aplicación este construida correctamente. El r el interrogante a responder es: ¿Se está construyendo la aplicación de manera correcta?

La Verificación es el proceso de determinar cuando o no, los artefactos de una fase o etapa del desarrollo del software, cumplen los requerimientos establecidos en la fase anterior.[1]

La Verificación implica revisiones y reuniones para evaluar documentos, planes, código, requerimientos, especificaciones, para confirmar la conformidad o cumplimiento de los requerimientos. Utiliza técnicas como las pruebas, inspecciones y revisiones.

La Validación se entiende como determinar la exactitud del producto de software con respecto a las necesidades y requisitos del usuario; se trata de responder la inquietud: ¿Se construyó la aplicación correcta? La Validación permite al desarrollador del software asegurar que el sistema o aplicación satisface las expectativas del cliente. Las técnicas utilizadas, son las pruebas, inspecciones y revisiones.

La V & V(Verificación y Validación) se enfocan hacia la determinación de correcciones y cumplimiento de los propósitos.

1.2 Enfoques de Verificación y Validación

Existen cuatro paradigmas que permiten la aplicación de las actividades de V&V en un proyecto de software [2]: Análisis manual, estático, dinámico y formal.

El Análisis Manual se realiza mediante revisión de los requerimientos del software con la ayuda de un experto/expertos en el dominio del problema que se va a solucionar. Entre las desventajas de este primer enfoque, está el costo (de los expertos a contratar), no hay garantía que se cumplan todas las exigencias del proceso de V&V para la fase de requerimientos, en el caso de que sea realizado el análisis por otro grupo se podría presentar cierto grado de subjetividad en los resultados. El Análisis Manual puede ser difícil de realizar de acuerdo a la complejidad del dominio del problema a modelar.

El Análisis Estático involucra la ayuda de una herramienta que busca analizar la representación estática del sistema (diagramas de flujo, casos de uso, código fuente, otros) con el fin de descubrir errores. Este análisis implica la no ejecución de modelos ni del código fuente para determinar las posibles fallas. Se trata por lo tanto de verificar los

“productos” obtenidos durante el desarrollo del software. El Análisis Estático involucra técnicas como la inspección, chequeos de especificaciones, diseños y código, análisis automatizado de programas, normalmente mediante flujo de datos y verificación de estados.

El Análisis Estático normalmente no es un reemplazo de las pruebas de software, sino que las complementa, enfocándose en aquellas no conformidades que no son posibles detectar usando únicamente pruebas.

El Análisis Dinámico implica normalmente la ejecución del código fuente del software con datos de prueba, con el fin de probar y evaluar en tiempo real los diferentes escenarios soportados por el programa.

El Análisis Formal, implica representar matemáticamente el modelo de los requerimientos, con el fin de realizar posteriormente el análisis para determinar el cumplimiento del modelo. Implica por lo tanto un Análisis Estático, pero desde el punto de vista formal.

Para el caso de la herramienta GERESOF, el tipo de Análisis implementado es Estático, puesto que hace uso de una de las técnicas utilizadas para este fin, como lo son las Revisiones.

1.3 Ingeniería de Requisitos

Es el proceso de descubrir, analizar, documentar y verificar los servicios proporcionados por el sistema y sus restricciones operativas. Los servicios corresponden a las necesidades de los clientes que son satisfechas por la aplicación [2]. La Ingeniería de Requisitos también puede ser vista como una actividad de “Ingeniería” y “Gestión”; desde la primera,

porque es concerniente con la identificación de metodologías apropiadas para desarrollar soluciones de software bajo unos costos que sean apropiados para su implementación; desde la segunda, porque mediante actividades de gestión como la monitorización de los requerimientos del producto y el alcance del proyecto, los costos y la agenda a través del proceso de desarrollo de software, se asegura que todas las aplicaciones del negocio sean entregadas como se definen en los documentos de especificación de requerimientos[4].

Desde diferentes enfoques, es claro que la Ingeniería de Requisitos es una de las fases más cruciales del diseño y desarrollo de software, debido a que esta soporta el problema crítico de estar diseñando el producto/aplicación correcta para el cliente. Teniendo en cuenta este importante argumento, la herramienta GERESOF fue concebida para apoyar la Ingeniería de Requisitos, desde el punto de vista de la Gestión, involucrando los procesos de Verificación y Validación.

Para esto, fue necesario explorar y evaluar los diferentes modelos de procesos de Ingeniería de Requisitos, con el fin de seleccionar el que el mejor se acercaba a la funcionalidad de la herramienta propuesta.

1.4 Modelo de Gestión de Requisitos

Una de las empresas más importantes y experimentadas en términos del Desarrollo de Software, es la BORLAND (The Open Alm Company), en un documento realizado por el señor Jordi Borja el día 20 de Junio de 2008, Director General Borland Ibérica, menciona que un punto clave a la hora de desarrollar software es MEJORAR LOS PROCESOS pensando en que la herramienta desarrollada NO LO

ES TODO; para lograrlo hay que mejorarlo desde los subprocesos que componen la Definición y Gestión de Requisitos, por ello presenta el siguiente modelo (ver Figura 1) donde están relacionados los aspectos que deben cumplir y revisar a la hora de realizar estas actividades:

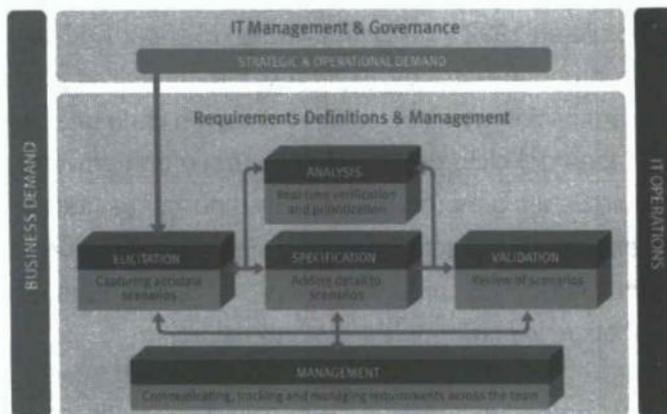


Figura 1. Modelo de Subprocesos de Definición y Gestión de Requisitos Básicos.

Fuente: Tomado de [http://www.slideshare.com/...](http://www.slideshare.com/) Consultado el 28 de Junio de 2008.

El propósito que fundamentó esta investigación fue generar una solución basada en software libre que permitiera mejorar el proceso de Verificación y Validación; la orientación que se dio hacia el tema de la Gestión de Requisitos resultó al comprender los referentes encontrados en múltiples empresas y grupos donde enfatizan en herramientas que permitan sistematizar estos procesos con el fin de obtener una mejor Calidad del Software implementado.

1.5 Procesos de la Gestión de Requisitos

Teniendo en cuenta el Modelo de Gestión de Requisitos seleccionado, es necesario conocer los procesos que están involucrados en el mismo:

A. ELICITACIÓN [5]: Entendiéndose ésta como el proceso de descubrir el contexto en que se desarrolla el software, tornándolo lo más explícito posible y obteniendo el máximo de información para el conocimiento del objeto en cuestión, en este caso los requisitos.

Entre las actividades y tareas involucradas se tienen:

- Selección de Técnicas de Elicitación
- Identificación de Stakeholders
- Definición de Escenarios por el Usuario
- Establecimiento de límites del Sistema
- Gestión de Dependencias
- Creación de un Glosario y del Modelo de Dominio

B. ESPECIFICACIÓN: Mediante este proceso se formalizan (definen, analizan y verifican) los requisitos de sistema de una aplicación, a partir de los requisitos de los usuarios, objetivos de negocio, restricciones de diseño, estándares externos, entre otros.

Entre las actividades y tareas involucradas se tienen:

- Detallar los escenarios de usuario
- Creación del Modelo de Casos de Uso
- Creación de prototipo de la interfaz de usuario
- Construcción / mejoramiento del modelo de negocio

C. ANÁLISIS: este proceso facilita la refinación de los requisitos, por lo que se encuentran entre sus actividades y tareas:

- Priorización de Alto nivel a escenarios de usuarios
- Verificación y Análisis

D. VALIDACIÓN: se realiza un seguimiento y se revisa el grado de cumplimiento con los requerimientos iniciales introducidos al sistema.

Entre las actividades y tareas se tiene:

- Revisión / Validación de Requisitos
- Inspecciones
- Pruebas

E. GESTION: involucra la administración de las fases anteriores, durante el proceso de gestión y definición de los requerimientos, facilitando la comunicación y seguimiento durante este proceso.

2. METODOLOGÍA

Para el desarrollo de la herramienta GERESOFT, se tuvieron en cuenta las siguientes estrategias metodológicas:

2.1 Proceso de Desarrollo Unificado

El Proceso de Desarrollo Unificado (RUP - Rational Unified Process), junto con el UML (Lenguaje Unificado de Modelado), los cuales constituyen la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos [5]. El RUP facilitó el desarrollo, mediante las iteraciones en las cuatro fases (ver Figura 2), con el apoyo de UML para modelar y especificar el sistema.

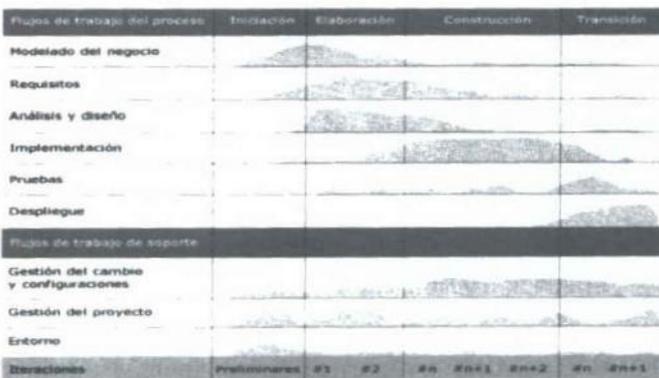


Figura 2. Proceso Unificado de Desarrollo.

Tomado de http://es.wikipedia.org/wiki/Imagen:Rup_espanol.gif

2.2 Framework de la IEEE

Para el desarrollo del prototipo se tuvo en cuenta uno de los patrones más generalizados dentro de los referentes conocidos en cuanto a los modelos de verificación y validación existentes, es decir, el de la IEEE 1012-1986/1992/2004. Este estándar fue revisado en 1992 y por última vez en el año 2004. En el se propone entre otras cosas, que las actividades de verificación y validación se realicen en cada una de las fases del ciclo de vida de desarrollo de software, que se deben generar unos informes de los resultados obtenidos en las actividades de V & V (Verificación y Validación) y los procedimientos a realizar para el control de no conformidades o anomalías.

De acuerdo al estándar [7], 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).
- Satisfacen las normas, prácticas y convenciones en los procesos del ciclo de vida.
- Completan exitosamente cada una de las actividades del ciclo de vida.

Y 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.
- Resuelven el problema correctamente.

Una primera aproximación a la organización del estándar, se muestra en la figura 3:

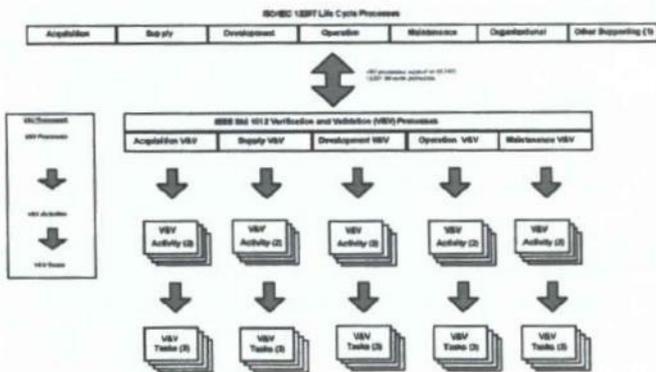


Figura 3. Organización del Estándar IEEE 1012.

De la Figura 3, destacamos el modelo que se implementó en la herramienta GERESOFT, el cual se hace énfasis en los procesos, actividades y tareas, como se muestra en la figura 4:

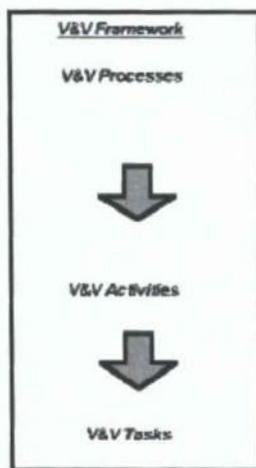


Figura 4. Framework de V&V del Estándar IEEE 1012

Los Procesos o también llamados Subprocesos dentro de la Gestión de Requisitos, tienen asociados a ellos el cumplimiento de Actividades, y se referencian a través de atributos complementarios que acompañan a las Tareas a Verificar y Validar (V&V),

es por ello que al utilizar un framework que generalice los procedimientos se debe definir los nombres de los procesos, actividades y tareas en orden jerárquico de dependencia, permitiendo una mayor personalización y flexibilidad al decidir cuántos serán y cómo se llamarán, obviamente guardando la relación de orden mencionada anteriormente.

3. RESULTADOS

El producto generado con este trabajo dio como resultado un Software denominado GERESOFT por sus autores, el cual corresponde a "Gestión de Requerimientos de Software".

Teniendo en cuenta que el proyecto se desarrolló en la ciudad de Cartagena de Indias, ubicada en la Costa Caribe colombiana, se buscó un elemento complementario y representativo de la región, por lo que se seleccionó el pájaro conocido como "María Mulata", con el cual el logo definitivo de la aplicación se muestra en la Figura 5:



Figura 5. Logo Geresoftware

3.1 Modelo del Negocio de GERESOFT

Identificación de Requisitos: Con la identificación de requisitos se realizó un análisis profundo acerca de los diferentes modelos de verificación y validación existentes, además de que se obtuvo un estudio comparativo de las herramientas de Verificación y Validación de gestión de requisitos.

Se definió la utilización del esquema de verificación y validación de la IEEE, operacionalizando esta labor a

través de procesos, actividades y tareas, citadas en el documento.

Además, se obtuvieron los requerimientos funcionales de la solución basada en software libre a partir del estudio comparativo mencionado.

Requerimientos Funcionales del Software: Durante la investigación para el diseño de este proyecto, se formularon los casos de uso de alto nivel y posteriormente en el nivel más detallado, como se muestra en la figura 6 :

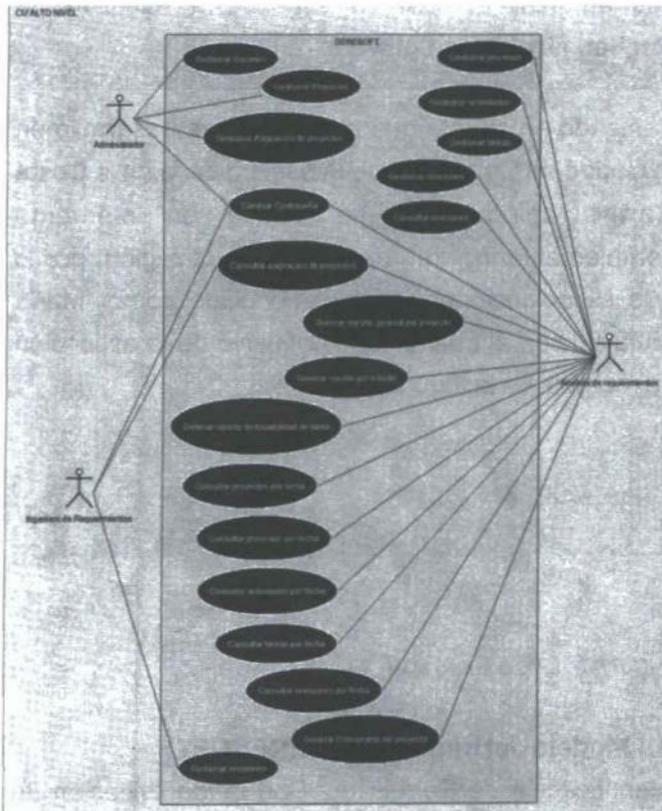


Figura 6. Casos de Uso de Alto Nivel de GERESOFT

Estos casos de uso se construyeron a partir de los casos preliminares y del análisis de cada uno de ellos y se adhieren estrictamente al diseño del programa, mostrando el funcionamiento del mismo paso a paso.

En términos generales se puede dividir este sistema en dos Subsistemas:

- **Subsistema Administrativo:** En este subsistema se deben Gestionar los Usuarios y Proyectos del sistema y Asignar los Usuarios y perfiles a los Proyectos. El actor que tiene los derechos para acceder a este sistema es solo el Administrador.
- **Subsistema Operativo:** En esta parte del Sistema se Gestionan los Procesos, Actividades, Tareas y Relaciones creadas por el Analista de Requerimientos, y las revisiones de V&V son ejecutadas por el Ingeniero de Requerimientos quien es el encargado de efectuar dichas labores en este subsistema. El Analista de Requerimientos puede adicionalmente Gestionar Reportes sobre el proceso de V&V, la trazabilidad y revisar el estado en que se encuentra dicho proyecto.

Igualmente los Actores o grupos de usuarios que interactúan en el software se pueden clasificar de la siguiente manera:

- **Administrador:** Quien crea los usuarios del sistema, proyectos y los asigna a los mismos teniendo en cuenta los roles que tienen ellos.
- **Analista de Requerimientos:** Quien asigna los Procesos, Actividades, Tareas, Relaciones, Revisiones que deben atender los Ingenieros de Requerimientos. Gestiona la generación de informes a partir de la información más relevante para conocer el estado de Verificación y Validación proyecto.
- **Ingeniero de Requerimientos:** Es quien ejecuta la revisión de las relaciones entre tareas

produciendo la información necesaria para complementar la Verificación y Validación de las relaciones aplicadas a la Gestión de Requisitos a través del Framework utilizado.

3.2 Arquitectura GERESoft

La herramienta responde a un modelo de capas o niveles, el cual permite balancear adecuadamente el uso de los recursos y las funcionalidades:

- Cliente soportado en un navegador web, con tecnologías HTML/XHTML, CSS, Java Script y Ajax.
- Lógica del Negocio soportada en el uso de clases mediante el lenguaje de programación web PHP, con el soporte del servidor de aplicaciones Apache HTTP Server.
- Base de Datos soportada en la base de datos MySQL

Gráficamente el modelo responde a lo mostrado en la figura 7:

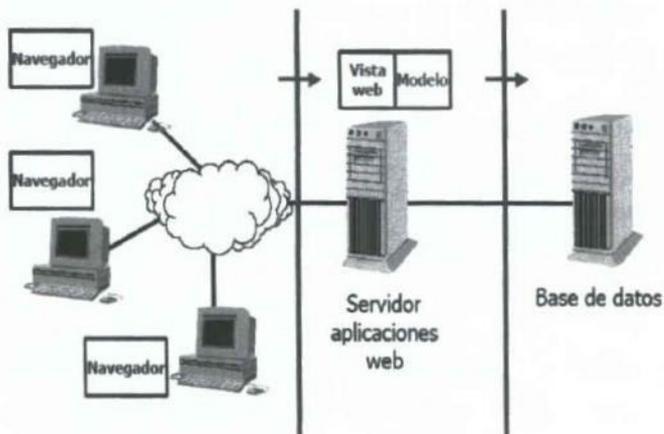


Figura 7. Arquitectura General de GERESoft

3.3 Estrategia de Diseño y Desarrollo del Software

El software desarrollado sirve para aplicar los procesos de Verificación y Validación durante la fase de Gestión de requisitos en el Diseño y Desarrollo de un proyecto, por este motivo para explicar la funcionalidad hay que señalar que la clasificación de la información se basó en el Framework de la IEEE y a través de éste se estructuraron las operaciones a realizar y el alcance definiendo los procesos, actividades y tareas que se realizan en esta parte del ciclo de software. Una lista jerárquica de este dominio a seguir para enumerar los Procesos, Actividades y Tareas podría ser la mostrada en la Tablas 1, 2 y 3:

Tabla 1. Listado de Procesos

PROCESOS	Elicitación
	Especificación
	Análisis
	Validación
	Gestión

Tabla 2. Listado de Actividades

ACTIVIDADES	1. Realización de Entrevistas
	2. Realización de encuestas
	3. Realización de otros instrumentos de recolección de información.
	4. Otras actividades de este proceso
	1. Documentación de especificación y Detalle de Escenarios
	1. Verificación en tiempo real.
	2. Priorización de requerimientos
	3. Verificación en tiempo real.
	4. Priorización de requerimientos
	1. Revisión de Escenarios
1. Almacenamiento de información.	
2. Cambios en los requerimientos.	
3. Cruce de información entre el equipo.	

Tabla 3. Listado de Tareas

TAREAS	<ol style="list-style-type: none"> 1. Obtener información sobre el dominio del problema y el sistema actual. 2. Preparar y realizar las reuniones de elicitación. 3. Identificar / revisar los objetivos del sistema. 4. Identificar / revisar los requerimientos de almacenamiento de información. 5. Identificar / revisar los requisitos funcionales. 6. Identificar / revisar los requisitos no funcionales. 7. Priorizar objetivos y requisitos
	<ol style="list-style-type: none"> 1. Descripción de los requerimientos funcionales 2. Descripción de los requerimientos no funcionales 3. Descripción de las características de diseño 4. Modelado de los requerimientos mediante artefacto seleccionado (Diagramas de Flujo, Casos de Uso, otros).
	<ol style="list-style-type: none"> 1. Se priorizan los requerimientos modelados en los artefactos del sistema. 2. Se genera el documento de la arquitectura del sistema y de las diferentes vistas del mismo.
	<ol style="list-style-type: none"> 1. Trazabilidad de los requerimientos 2. Evaluación de los requerimientos 3. Generación del plan de pruebas.
	<ol style="list-style-type: none"> 1. Definición de los requerimientos del usuario, del sistema. 2. Gestión de los Stakeholders. 3. Gestión de la planeación de requerimientos. 4. Gestión de la información y Configuración. 5. Gestión de la Aseguramiento de la Calidad.

Con este diseño y desarrollo, el software permite configurar cuántos y cómo se llamarán los diferentes niveles del Framework de Verificación y Validación utilizado, brindando una cualidad importante que es la estandarización y adecuación a futuras adaptaciones a partir de la misma jerarquía señalada.

3.4 Procesos para el manejo del software

Luego de comprender la estrategia de diseño y desarrollo que se utilizó para realizar la Verificación y Validación con este software, se presenta la secuencia de procesos para el manejo de la herramienta, como se muestra en la siguiente Figura 8:

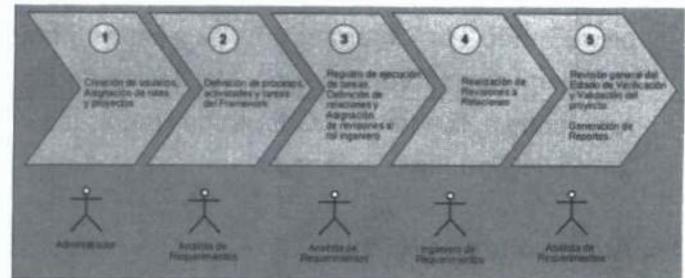


Figura 8. Secuencia de procesos para el manejo de la Aplicación.

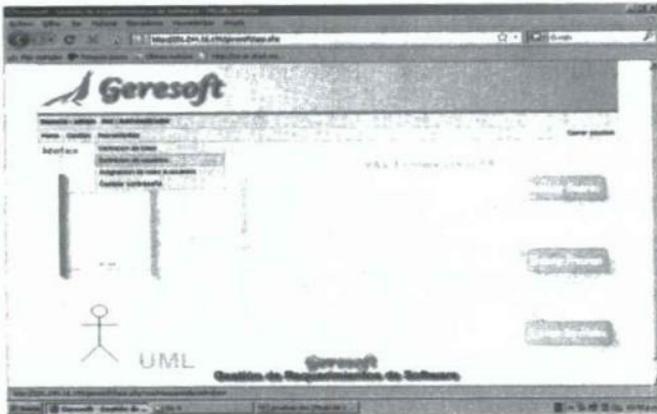
Primeramente, el administrador de la aplicación ha realizado la creación de usuarios, asignándoles roles en cada proyecto con los cuales estén relacionados (Ver figura 8. Secuencia de procesos para el manejo de la Aplicación, Paso 1).

Luego el Paso 2 de la Figura 8, muestra que el Analista de Requerimientos es quien define los procesos, actividades y tareas que se van a realizar.

Seguidamente, cuando las tareas ya son ejecutadas se hace necesario registrar su ejecución y subir el formato que evidencia este desarrollo (Ver paso 3 de la Figura 8). Esto se realiza actualizando los datos de la ejecución y el registro de plantillas que evidencian el desarrollo de la tarea, éstas son relacionadas y programadas para su revisión por parte de los ingenieros de requerimientos.

Después en el Paso 4 de la Figura 8, el Ingeniero de Requerimientos realiza las revisiones sobre las relaciones creadas anteriormente, dejando sus comentarios y actualizando el estado de Verificación y Validación.

Finalmente, el Analista de requerimientos puede realizar una Revisión General del estado de Verificación y Validación del proyecto y la generación de reportes con la información relacionada a la V&V, como se aprecia en el Paso 5 de la Figura No. 8.



3.5 Vistas de la Aplicación

Se pueden apreciar en el subsistema administrativo diversas vistas para definir los usuarios, roles, proyectos y asignación de roles a los mismos, entre ellas se pueden apreciar las siguientes interfaces de usuario (Figuras 9 y 10):



Figura 9. Menú Definición de Usuarios.

Figura 10. Asignación de Rol a Usuarios. Y para el subsistema operacional con el cual se configuran los niveles del Framework para realizar la Verificación y la Validación a través de la jerarquía propuesta (Los Procesos, Actividades y Tareas) se encuentran las siguientes interfaces gráficas del software GERESOFT (ver Figuras 11 al 14):

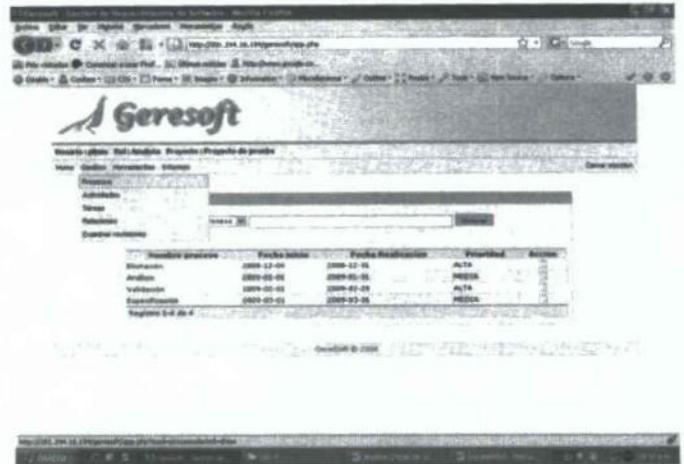


Figura 11. Menú Procesos

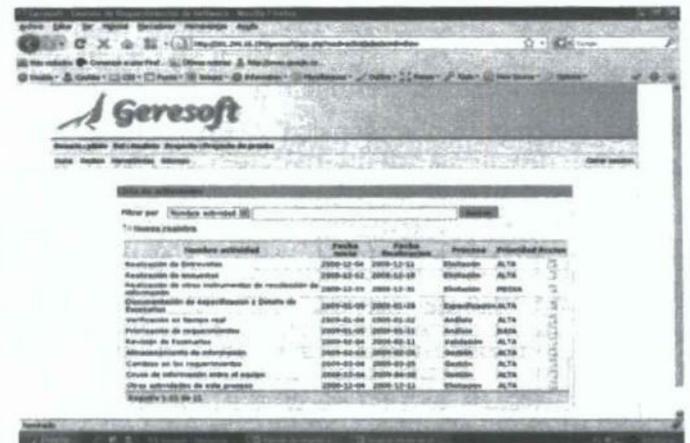


Figura 12. Lista de Actividades



Figura 13. Nueva Tarea

