



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 3

Centro para la Informatización de la Gestión de Entidades

**PROCESO DE DESARROLLO DE REQUISITOS PARA SISTEMAS
INTEGRALES, MODULARES Y ADAPTABLES APLICADO AL
PROYECTO ERP-CUBA**

**Trabajo final presentado en opción al título de
Máster en Informática Aplicada**

Autora: Ing. Tahirí Rivero Alvarez

Tutores: MSc. Yanet Vega Miniet

MSc. Sasha Valdés Jiménez

Dra. Ailyn Febles Estrada

La Habana

Diciembre de 2012

AGRADECIMIENTOS

Deseo agradecer a los tutores por sus acertadas observaciones y la paciencia durante los largos debates realizados.

Un agradecimiento especial a Larisa González Álvarez por las no pocas horas dedicadas a reflexionar sobre la solución que aquí se expone y por el ánimo que siempre me propició cuando más falta me hacía.

Agradecer también al equipo de analistas del proyecto ERP-Cuba por su colaboración en la aplicación de esta propuesta y especialmente a la analista principal Tamara Rodríguez Sánchez por su trabajo incondicional en cualquier horario.

Agradecer a todos los que me hicieron críticas constructivas en los talleres de tesis, en la calle, etc. A todos los que se preocuparon por mi avance.

A mis compañeros de trabajo por las concesiones que tuvieron conmigo con respecto a los horarios y por asumir en muchas ocasiones mis tareas para que yo pudiera avanzar.

Agradecer a mi familia por seguir paso a paso el desarrollo de esta tesis.

- A mi tía Quilly por empujarme y no dejar que desvíe mi centro de atención.
- A mi papi, mis hermanos y mi abuelita porque con sus deseos me han hecho tomar esto como una meta de obligado cumplimiento.
- A mi Rafe (mi futuro esposo) por seguir el avance de cada párrafo de esta tesis. Muchas gracias por la fuerza y la tranquilidad que me propicias. Sin ti mis nervios nunca hubieran resistido el estrés por el que he pasado con esto.

A todos los que aportaron de alguna manera su granito de arena decirle MUCHAS GRACIAS. Al fin “LO LOGRAMOS”.

DECLARACIÓN JURADA DE AUTORÍA

Declaro por este medio que yo **Tahirí Rivero Alvarez**, con carné de identidad **85031415839**, soy la autora del trabajo final de maestría **“Proceso de Desarrollo de Requisitos para sistemas integrales, modulares, y adaptables aplicado al proyecto ERP-Cuba”**. Finalmente asumo la responsabilidad moral y jurídica que se derive de este juramento profesional.

Y para que así conste, firmo la presente declaración jurada de autoría en La Habana a los ____ días del mes de _____ del año _____.

Ing. Tahirí Rivero Alvarez

MSc. Yanet Vega Miniet

MSc. Sasha Valdés Jiménez

Dra. Ailyn Febles Estrada

DEDICATORIA

En la memoria de mi mami, que en estos momentos sería la persona más feliz del mundo. Esto es por ti y para ti donde estés.

De tu niña

RESUMEN

La Ingeniería de Requisitos es una etapa crítica en el desarrollo de software ya que el resultado de este proceso guía la futura implementación del sistema. Dentro de esta disciplina se encuentra el Desarrollo de Requisitos, el cual presenta mayor interés cuando se debe garantizar desde etapas tempranas la modularidad y adaptabilidad de los sistemas que tienen estas características como distintivas. En esta investigación se realiza un proceso para llevar a cabo el Desarrollo de Requisitos ajustado a los elementos anteriormente mencionados y fomentando la reutilización de los mismos. Para esto se modifica el proceso tradicional de Desarrollo de Requisitos, incluyéndole nuevas actividades y artefactos. Se utilizan como base para la modificación los aspectos positivos observados durante la revisión bibliográfica en trabajos relacionados con la temática que aquí se aborda. Con la aplicación de esta propuesta en el proyecto ERP-Cuba se demuestra un aumento en el grado de adaptabilidad y modularidad que puede lograrse a nivel de requisitos en la construcción de sistemas con estas características.

Palabras claves: Ingeniería de Requisitos, modularidad y adaptabilidad

ABSTRACT

Requirements Engineering is a critical stage in the development of software as the result of this process guide the future implementation of the system. Within this discipline is the development of requirements, which presents greater interest when due from an early stage to ensure the modularity and adaptability of systems that have these features as distinctive. In this investigation, a process is performed to carry out development requirements set to the abovementioned elements and promoting the reuse thereof. To this traditional process amending Requirements Development, including you new activities and artifacts. They are used as a basis for modifying the positive aspects observed during the literature review on work related to the topic addressed here. With the implementation of this proposal in the ERP-Cuba project demonstrated an increase in the degree of adaptability and modularity can be achieved at the level of requirements in building systems with these characteristics.

Keywords: *Requirements engineering, modularity and adaptability.*

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTO TEÓRICO	6
LOS REQUISITOS DE SOFTWARE	6
LA INGENIERÍA DE REQUISITOS Y SUS ACTIVIDADES	8
DESARROLLO DE REQUISITOS PARA SISTEMAS MODULARES, INTEGRALES Y ADAPTABLES	13
TRABAJOS RELACIONADOS CON EL DESARROLLO DE REQUISITOS PARA SISTEMAS INTEGRALES, MODULARES Y ADAPTABLES.	18
CONCLUSIONES PARCIALES	24
CAPÍTULO 2 DESARROLLO DE REQUISITOS PARA SISTEMAS INTEGRALES, MODULARES Y ADAPTABLES	25
CONSIDERACIONES GENERALES DE LA PROPUESTA	25
MODELADO Y DESCRIPCIÓN DEL PROCESO DE DESARROLLO DE REQUISITOS PARA SISTEMAS MODULARES, INTEGRALES Y ADAPTABLES	37
CONCLUSIONES PARCIALES	48
CAPÍTULO 3 VALIDACIÓN DE LA PROPUESTA	49
POBLACIÓN Y SELECCIÓN DE LA MUESTRA.....	49
EJECUCIÓN DEL EXPERIMENTO	55
IMPACTO SOCIAL	63
CONCLUSIONES PARCIALES	63
CONCLUSIONES GENERALES	64
RECOMENDACIONES	65
REFERENCIAS BIBLIOGRÁFICAS	66
ANEXOS	69

INTRODUCCIÓN

La creciente necesidad de las empresas de organizar y controlar sus recursos ha traído consigo que estas opten por la inclusión de sistemas informáticos como es el caso de los Sistemas de Planificación de Recursos Empresariales (ERP, por sus siglas en inglés). Estos sistemas son considerados mercados prometedores y de vital importancia para las empresas debido a su capacidad de integración de procesos y adaptación de estos a distintos entornos[1], [2]. Se definen por Estevez como "...un paquete software, compuesto por múltiples módulos, que ofrece soluciones integradas diseñadas para dar soporte a múltiples procesos de negocio (recursos humanos, ventas, finanzas, producción, entre otros), proporcionando una integración de los datos de la organización con los procesos de negocio" [3].

Una de las metas que tienen las empresas desarrolladoras de este tipo de sistema es construir soluciones a la medida partiendo de otras previamente elaboradas. Esto se debe a dos razones fundamentales:

- Las características de modularidad, integralidad y adaptabilidad que presentan este tipo de sistemas.
- Todas las empresas para llevar a cabo su control interno utilizan uno o varios de sus módulos, ya que presentan áreas funcionales similares.

Según Carreón la característica de integralidad está asociada a permitir controlar los diferentes procesos de la organización y sus relaciones, por lo que los datos que se ingresan sólo una vez en uno de los módulos deben poder ser empleados por los restantes. La característica de modularidad está dada por la capacidad que tenga el sistema de desacoplarse para instalar o no los módulos dependiendo de las solicitudes de los clientes y la característica de adaptabilidad está dada por la capacidad que tenga el sistema de adecuarse a los requisitos específicos de la empresa [4]. Esta última característica según Lage y Carreón es el valor añadido fundamental con el que deben contar los ERP[4], [5].

El Ministerio de Finanzas y Precios de Cuba, en la búsqueda del fortalecimiento de la gestión de entidades, como parte de la estrategia de informatización de la sociedad, puso en marcha en el 2008 el proyecto denominado ERP-Cuba, con el desarrollo del producto Cedrux como solución informática. "Con la creación de este producto se cuenta con una solución nacional que permite, a diferencia de otros productos

similares, la gestión integral de las entidades presupuestadas y empresariales, basada en los principios de independencia tecnológica”[6].

Este sistema integral se desarrolla por el Centro para la Informatización de la Gestión de Entidades (CEIGE) perteneciente a la Universidad de las Ciencias Informáticas, el cual tiene la misión de obtener un producto que pueda ser implantado en diferentes entornos empresariales (de producción y/o de servicios) y ser comercializable fuera del país. Una vez concluida una primera versión del producto se comenzaron a desarrollar soluciones específicas para distintos clientes. A partir de este momento comenzaron a surgir algunas problemáticas referentes a las características distintivas del sistema. En esta investigación se tratarán solamente las problemáticas referentes a la modularidad y la adaptabilidad del sistema, las mismas fueron:

- Se pudo formar solo algunos paquetes de soluciones, debido a que los productos desarrollados presentaban un alto acoplamiento funcional. Este acoplamiento limitó la creación de múltiples soluciones.
- Durante un piloto realizado se identificaron funcionalidades que podían haber sido cubiertas por el sistema si hubiese contado con un mayor grado de parametrización o configuración.
- Se desarrollaron productos desde cero, que aunque no podían ser cubiertos en su totalidad por las parametrizaciones debido a las particularidades de los negocios, podían reutilizar algunas funcionalidades de los existentes. Esto trajo consigo que se empleara tiempo en realizar análisis de requisitos que ya se encontraban especificados en otro producto.

“La Ingeniería de Requisitos es el proceso de desarrollar una especificación de software. Las especificaciones pretenden comunicar las necesidades del cliente a los desarrolladores del sistema”[7]. Algunos autores plantean que esta puede dividirse en dos momentos, uno donde se lleva a cabo el Desarrollo de Requisitos y otro donde se lleva a cabo la Administración de los mismos [8],[9].

Varios autores como Pressman en [9], Soto en [10], Valdés en [11] coinciden en que la Ingeniería de Requisitos es crítica en el desarrollo del software, por lo que se puede afirmar que los errores cometidos en esta etapa durante la construcción de los productos que integraron CedruX y durante la personalización de otras soluciones, tributaron a los errores anteriormente mencionados. Aunque es válido señalar que en

algunos productos desarrollados estos errores se comportaron en menor grado que en otros, por haber realizado de forma empírica algunas acciones en función de lograr cubrir las necesidades del ERP-Cuba, los errores cometidos de manera general son:

- No se orientó el Desarrollo de Requisitos a contribuir con el desacoplamiento de las funcionalidades.
- No se orientó el Desarrollo de Requisitos a contribuir con que los requisitos pudieran tener mayor grado de adaptabilidad a distintos entornos.
- No se orientó el Desarrollo de Requisitos a que se fomentara la reutilización de estos entre productos.

A partir de esto se plantea como **problema** a resolver:

¿Cómo llevar a cabo el Desarrollo de Requisitos ajustado a las necesidades del proyecto ERP-Cuba, de manera que se contribuya a aumentar el grado adaptabilidad y modularidad de los productos *en la etapa de Ingeniería de Requisitos?*

Objeto de estudio

Disciplina de Ingeniería de Requisitos.

Objetivo

Obtener un proceso para el Desarrollo de Requisitos ajustado a las necesidades del proyecto ERP-Cuba que contribuya a aumentar el grado de adaptabilidad y modularidad de los productos *en la etapa de Ingeniería de Requisitos.*

Objetivos específicos

- Establecer un marco teórico relativo al Desarrollo de Requisitos y sus enfoques a la modularidad y adaptabilidad para definir los fundamentos teóricos de la investigación.
- Definir un proceso para llevar a cabo el Desarrollo de Requisitos ajustado a las necesidades del ERP-Cuba.
- Validar la propuesta mediante su aplicación en el proyecto ERP-Cuba.

Campo de acción

Proceso de Desarrollo de Requisitos en el proyecto ERP-Cuba.

Hipótesis

Si se emplea un proceso para el Desarrollo de Requisitos ajustado a las necesidades del proyecto ERP-Cuba aumentará el grado de adaptabilidad y modularidad de los productos en la etapa de Ingeniería de Requisitos.

Tareas de la investigación

- Resumir el estado de las tendencias actuales sobre el Desarrollo de Requisitos para lograr modularidad y adaptabilidad.
- Evaluar las tendencias seleccionadas y tomar una posición al respecto.
- Realizar modificaciones al flujo tradicional del Desarrollo de Requisitos para ajustarlo a las necesidades del ERP-Cuba.
- Definir actividades, roles, y artefactos que se deben ejecutar para apoyar las modificaciones propuestas.
- Aplicar las modificaciones propuestas al proceso de Desarrollo de Requisitos en productos del ERP-Cuba.
- Evaluar la propuesta a través de las variables de la investigación.

Métodos

Para apoyar la presente investigación se utilizan métodos teóricos y empíricos según el paradigma establecido en [\[12\]](#). Dentro de los métodos teóricos se utiliza el método Histórico Lógico para la revisión histórica de las tendencias existentes en el Desarrollo de Requisitos modulares y adaptables y el Analítico Sintético para sintetizar lo estudiado mediante los análisis correspondientes.

Se utiliza también el método Hipotético-Deductivo donde, partiendo del fundamento teórico realizado sobre los trabajos referentes a la investigación, se pueden plantear supuestos hipotéticos para realizar modificaciones al flujo de Desarrollo de Requisitos. Además mediante el ajuste de las necesidades del proyecto estudiado, se pueden obtener los indicadores deseados.

Como métodos empíricos se utilizó la Observación para caracterizar el entorno donde ocurre el problema. En este caso la observación fue del tipo incluida pues el investigador estaba presente en la observación de la problemática presentada relativa a la construcción de nuevos productos. Se utilizó además la Experimentación como mecanismo para verificar la veracidad de la propuesta y la Medición para cuantificar los

indicadores evaluados.

Novedad

Inclusión en el proceso tradicional de Desarrollo de Requisitos de:

- Análisis de modularidad y adaptabilidad de sistemas con estas características como distintivas.

Aporte práctico

- Proceso de Desarrollo de Requisitos ajustado a las características de los sistemas modulares integrales y adaptables.
- Artefactos que sustenten la propuesta de solución.
- Patrones para identificar y describir el comportamiento de requisitos típicos en sistemas de gestión.

Estructura del trabajo

El trabajo está estructurado en tres capítulos como se describe a continuación:

En el CAPÍTULO 1 se definen los principales conceptos que se tratarán. Se realiza un estudio del estado del arte sobre el enfoque de la Ingeniería de Requisitos para lograr modularidad y adaptabilidad de los sistemas que tienen estas características como distintivas.

En el CAPÍTULO 2 Se definen algunos elementos generales que sirven de base para el entendimiento de la propuesta. Se realiza el diseño de un proceso de Desarrollo de Requisitos ajustado a las necesidades de los productos con integralidad, adaptabilidad y modularidad. Se realiza una descripción detallada de cada una de las actividades y artefactos.

En el CAPÍTULO 3 se realiza la validación de la propuesta. Para esto se utilizan los productos del proyecto ERP-Cuba. Luego se hace un análisis de los resultados arrojados para llegar a una conclusión definitiva sobre la efectividad de la propuesta.

CAPÍTULO 1. FUNDAMENTO TEÓRICO

En este capítulo se realiza una revisión de las tendencias relacionadas con la Ingeniería de Requisitos para seleccionar el flujo básico por el que se regirá la propuesta. Se realiza además un estudio del estado del arte sobre el Desarrollo de Requisitos para sistemas adaptables, integrales y modulares, identificando los conceptos principales, las técnicas utilizadas y los trabajos existentes relacionados con esta temática.

Los requisitos de software

Según Brooks en [13] : “La parte más difícil de construir de un sistema software es decidir qué construir. [...] Ninguna otra parte del trabajo afecta más negativamente al sistema final si se realiza de manera incorrecta. Ninguna otra parte es más difícil de rectificar después.”

Desde el comienzo del desarrollo del software varios autores se han dado a la tarea de conceptualizar qué son los requisitos de software y cómo deben ser desarrollados para lograr el éxito final.

Según la IEEE¹ en [14] un requisito es:

- “(1) Condición o capacidad necesitada por el usuario para solucionar un problema o alcanzar un objetivo.
- (2) Condición o capacidad que debe cumplir o poseer el sistema o componente del sistema para satisfacer un contrato, estándar, especificación, u otro documento formalmente impuesto.
- (3) Representación documental de una condición o capacidad como en (1) o (2).”

Wieggers y Sommerville definen los requisitos de la siguiente forma: “Los requisitos son especificaciones de lo que debe ser implementado. Son descripciones de lo que debe hacer el sistema, o de propiedades o atributos que debe tener. Ellos pueden constituir restricciones al proceso de desarrollo del sistema” [8], [15]. Estas definiciones abordan los requisitos como expresión de la funcionalidad del sistema, como características no

¹ Instituto de Ingenieros Eléctricos y Electrónicos

funcionales que debe tener, por ejemplo: rendimiento, disponibilidad, integridad, y finalmente se refieren a que las restricciones impuestas al proceso de desarrollo también son requisitos. Sin embargo, no tratan a los requisitos como elementos de la calidad del producto ni tampoco se refieren a estos como puente de entendimiento entre los desarrolladores y el resto de los interesados en el proyecto para garantizar la satisfacción de los clientes y usuarios finales.

Teruel en [16] define a los requisitos de software como:

“... especificaciones de las características, cualidades, capacidades y comportamiento de un sistema, descritos de manera entendible para el equipo de desarrollo, clientes y usuarios y en función de la satisfacción de sus necesidades. Constituyen la base para la medición de la calidad del producto final.”

En esta definición se abordan los requisitos como puente de entendimiento entre todos los interesados para lograr la satisfacción de las necesidades de los clientes y usuarios finales y se identifican como base para mediciones de la calidad del producto final. Sin embargo, en ninguna de las definiciones analizadas se aborda que debe ser posible comprobar la presencia y correcta implementación del requisito en el producto o sistema desarrollado. En el marco del presente trabajo se define que un requisito es una representación de lo que debe hacer el sistema, o de las propiedades o atributos que debe tener para satisfacer las necesidades y expectativas de los clientes y usuarios finales, que sirve como puente de entendimiento entre estos y el equipo de desarrollo y cuya presencia y correcta implementación en el producto o sistema pueden ser comprobadas mediante un procedimiento finito.

Sobre los niveles que deben contener los requisitos varios autores como Sommerville en [15], Wiegers en [8], Pressman en [9] han realizado propuestas. En esta investigación se tratarán dos niveles de requisitos, los requisitos del cliente y los requisitos del producto. Los primeros se refieren a los procesos de negocio y los segundos a los tradicionales requisitos funcionales del producto. Esta posición es asumida para estar alineados con lo planteado en el Modelo de Capacidad y Madurez del Software (CMMI) en [17], bajo el cual se desarrollan los productos para los cuales va dirigida esta investigación.

La Ingeniería de Requisitos y sus actividades

Sommerville afronta la Ingeniería de Requisitos en cuatro actividades, donde se definen qué servicios requiere el sistema y las restricciones en el funcionamiento de este. La salida de este proceso es un documento de especificación de requisitos que se obtiene en dos niveles de detalle, uno de alto nivel para los usuarios y otro a un nivel más detallado para los programadores. Plantea además que este proceso debe realizarse en espiral para ir obteniendo de forma creciente las funcionalidades que contendrá el sistema. Para Sommerville el proceso de Ingeniería de Requisitos cuenta con las actividades que se muestra en la Figura 1 [7].

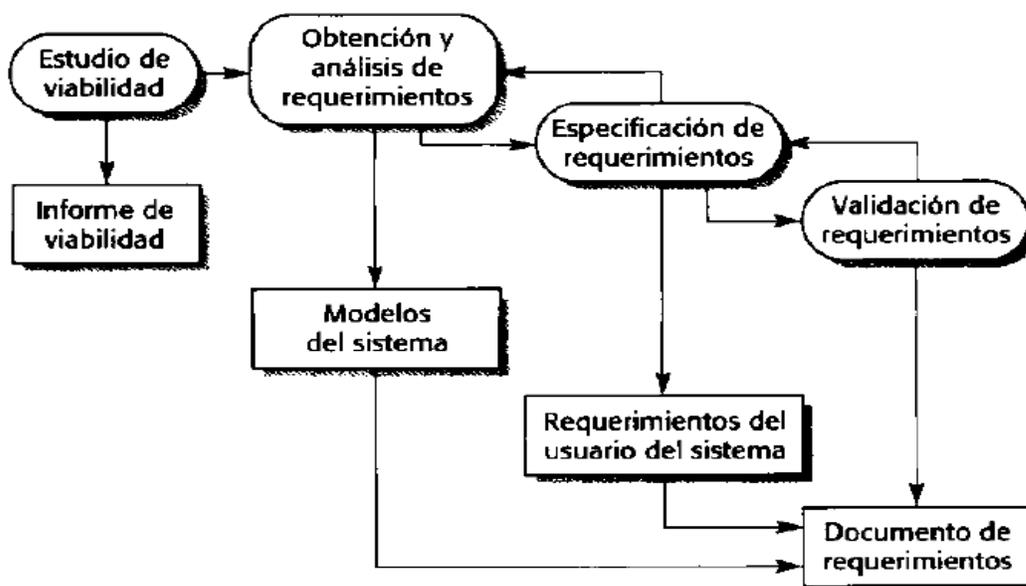


Figura 1 El proceso de Ingeniería de Requisitos [Tomado de Sommerville, 2005]

Wieggers por su parte plantea que es útil dividir la Ingeniería de Requisitos en dos partes, el desarrollo y la administración como se muestra en la Figura 2. La primera dedicada propiamente al desarrollo de las funcionalidades que contendrá el sistema y la segunda para llevar a cabo actividades de apoyo a la primera, como la gestión de cambios, el seguimiento de los requisitos, la gestión de proveedores, etc. Afirma que la ejecución de estos procesos debe realizarse en espiral, donde el proceso de administración de requisitos sirva como vía para comprender y controlar los cambios que se presenten mientras estén siendo desarrollados [8].

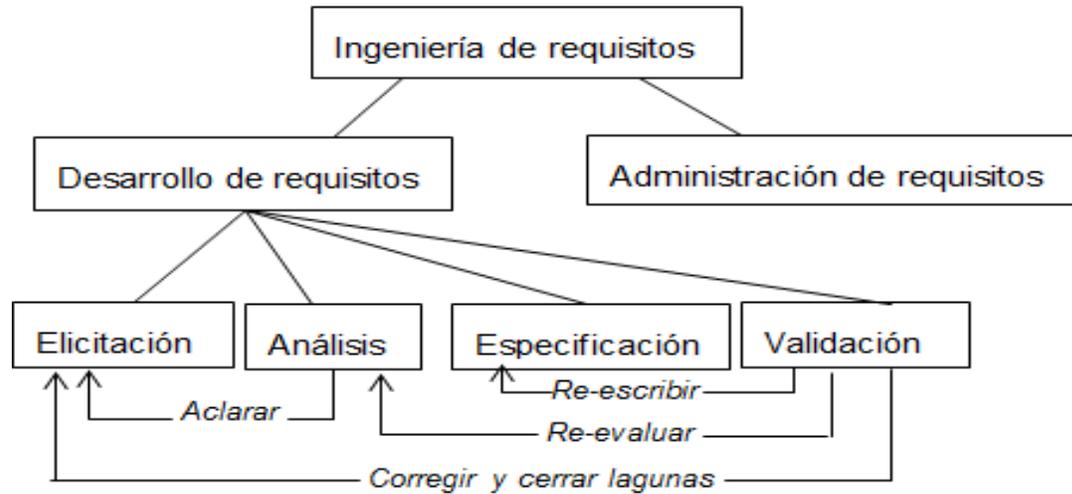


Figura 2 Etapas planteadas por Wieggers para la Ingeniería de Requisitos [Traducido de Wieggers, 2006]

Pressman en [9] coincide con Wieggers en [8] sobre las actividades que deben desarrollarse en la Ingeniería de Requisitos.

En [10] y [11] se realizan modificaciones al proceso de Ingeniería de Requisitos, con el objetivo de ajustarse a las condiciones del Sistema de Gestión Policial desarrollado en la Universidad de las Ciencias Informáticas bajo las buenas prácticas planteadas por CMMI y tomando como base lo planteado por Pressman y Wieggers sobre la Ingeniería de Requisitos. En la Figura 3 se muestra el resultado final de las modificaciones realizadas.

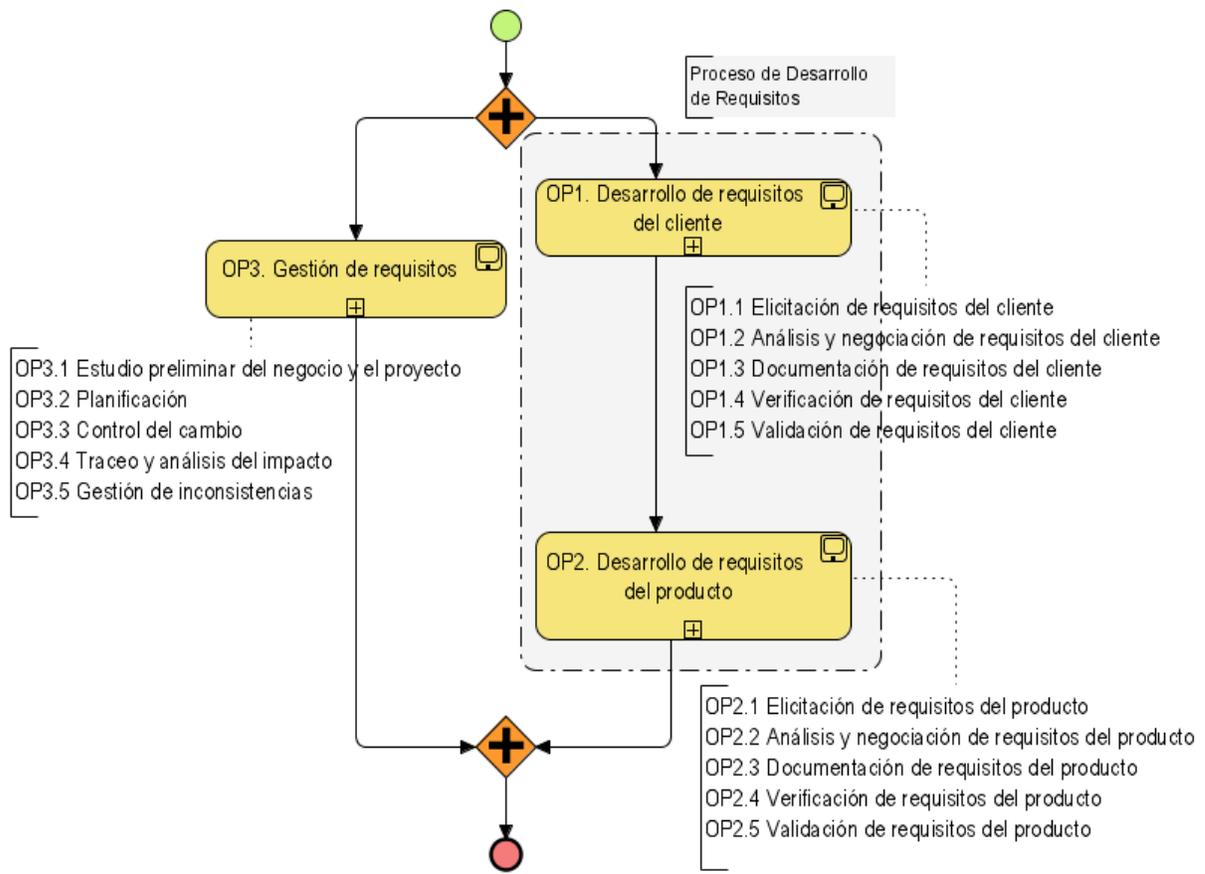


Figura 3 El proceso de Ingeniería de Requisitos [Tomado de Valdés, 2010]

Otra de las corrientes actuales estudiadas para llevar a cabo el Desarrollo de Requisitos es el enfoque dado en el desarrollo de líneas de productos de software. Según Díaz en [18] la definición sobre líneas de productos es la siguiente:

“Se definen las líneas del producto de software como un conjunto de sistemas software, que comparten un conjunto común de características, las cuales satisfacen las necesidades específicas de un dominio o segmento particular de mercado, y que se desarrollan a partir de un sistema común de activos base de una manera preestablecida”.

Las líneas de productos de software dividen el desarrollo de software en dos momentos el primero enfocado al desarrollo de elementos reutilizables, lo que también se le conoce como activos reutilizables y el segundo a desarrollar productos para clientes específicos partiendo de los activos anteriormente desarrollados. Esta forma de producción enfoca sus actividades en determinar aspectos comunes y variables en cada producto [18]. En la Figura 4 se muestra un flujo de actividades de este tipo de

desarrollo.

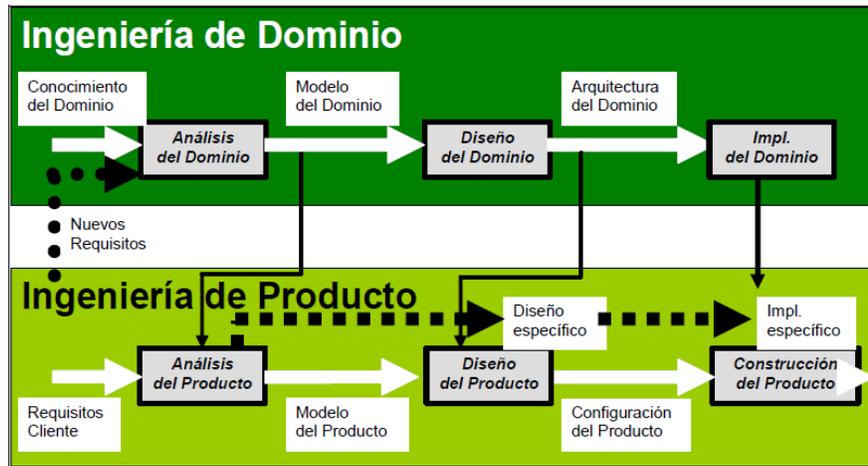


Figura 4 Desarrollo de líneas de productos de software [Tomado de Díaz, 2007]

Durante la Ingeniería de dominio se recogen iterativamente los requisitos comunes para toda la línea de productos de software. Una forma de expresarlos es mediante características que ayudan no sólo a que los clientes puedan distinguir unos productos de otros, sino que dirigen así mismo el desarrollo de código que implemente también esas características [18].

Según López y otros en [19] los dominios no son más que un área funcional diferenciable dentro de un contexto.

El SEI² plantea que un dominio es: Un área de conocimiento o actividad caracterizada por un conjunto de conceptos y la terminología entendida por los profesionales en esa área [20].

En la Ingeniería de producto se desarrollan requisitos para un cliente específico utilizando los requisitos previamente identificados en la ingeniería de dominio. El origen de esta técnica data de 1990 y fue propuesta por el Departamento de Defensa de los Estados Unidos [18].

Según Mellado en [21] y Rodríguez y otros en [22] “los requisitos de una línea de productos definen los productos de dicha línea y sus características comunes y variables, estos van a formar activos reutilizables para construir los productos de toda

² Instituto de Ingenieros del Software

la línea”. Según Montilva en [23] los activos son colecciones de partes de un software (requisitos, diseños, componentes, casos de prueba, etc.) que se configuran y componen para producir los productos de una línea de productos.

Valoraciones sobre las corrientes orientadas a la Ingeniería de Requisitos

En los estudios realizados sobre la Ingeniería de Requisitos se puede concluir que excepto Sommerville todos dividen la Ingeniería de Requisitos en dos etapas una de Administración y otra de Desarrollo de Requisitos. La primera como una etapa de apoyo a la segunda. Todos concuerdan en que en la etapa de desarrollo se deben realizar las siguientes actividades: Elicitación, Análisis y Negociación, Especificación y Validación como se muestra en la Tabla 1. Las etapas de la Ingeniería de Requisitos planteadas por estos autores no reconocen entre sus actividades el enfoque a la modularidad y adaptabilidad de los sistemas, por ser etapas genéricas para el desarrollo de cualquier sistema. Este trabajo se enmarcará en la etapa de Desarrollo de Requisitos para ajustarla a las necesidades de los sistemas modulares, integrales y adaptables.

Tabla 1 Comparación entre las corrientes de Ingeniería de Requisitos

Autores Etapas	Sommerville 2005	Wieggers 2006	Pressman 2010	Valdés 2010 Tesis UCI
Estudio de viabilidad	X			
Elicitación/Obtención	X	X	X	X
Análisis/Negociación	X	X	X	X
Especificación	X	X	X	X
Verificación				X
Validación	X	X	X	X
Administración		X	X	X

Desarrollo de Requisitos para sistemas modulares, integrales y adaptables

Desarrollo de Requisitos en sistemas adaptables

Según Carreón en [4] la adaptabilidad de un sistema está dada por la capacidad que tenga el mismo de ser configurable o parametrizable para distintos entornos empresariales. Con el objetivo de facilitar el entendimiento de esta característica es necesario precisar diferentes conceptos, tales como parámetro, parametrización y variantes de productos.

Según la Real Academia Española en [24] un parámetro es: “Un dato o valor que se toma como necesario para analizar o valorar una situación.”

Utilizando el concepto de parámetro y aplicándolo a un conjunto de ellos, se puede decir que una parametrización es la aplicación de un conjunto de parámetros a un contexto para analizar su comportamiento. Si este contexto son los sistemas informáticos se podría decir que es la aplicación de parámetros en un sistema informático que permita analizar las variantes de comportamiento que se producen una vez conjugados los parámetros. Si este concepto se aplica al contexto del ERP se puede afirmar lo siguiente:

Parametrización: Es la aplicación de un conjunto de parámetros a un sistema ERP que permita analizar las variantes de productos que se pueden construir una vez conjugados los parámetros y las reglas de negocio.

La parametrización es una de las características principales de este tipo de sistemas ya que les aporta un valor añadido al cubrir un gran número de dominios de aplicación existentes en el mercado empresarial. Durante la implantación de productos estas parametrizaciones se pueden realizar de dos formas, la primera mediante la configuración de los parámetros y flujos de trabajo y la segunda adaptando el producto mediante la modificación de código. Para esta investigación los productos que puedan ser parametrizados a partir de las configuraciones generales, se le llamará *aplicaciones parametrizables* y cuando la parametrización se realice mediante modificación del código ya sea para añadir o quitar requisitos de una aplicación existente se llamará *variante de producto*.

Para llevar a cabo el Desarrollo de Requisitos para este tipo de sistemas hay que tener en cuenta la parametrización que se desea realizar. En el primer caso, donde no se

necesita modificación de código, el Desarrollo de Requisitos debe estar orientado a garantizar que la aplicación que se construya contenga una gran cantidad de requisitos parametrizables, por lo que debe enfocarse a determinar partes comunes y variaciones en cada requisito. En el segundo caso donde se conforman variantes de productos hay que tener en cuenta que podrán reutilizarse requisitos. Autores como Sommerville en [7], y Mellado en [25] afirman que es necesario comenzar a pensar en la reutilización de software desde las primeras etapas de la construcción del mismo.

Según Ros en [26] el propósito de la reutilización de requisitos es identificar descripciones de sistemas que puedan ser reutilizadas (en su totalidad o en parte) de manera que se reduzca el esfuerzo en el desarrollo. La reutilización aunque es muy útil, es válido señalar que cuando una organización adopta un proceso basado en reutilización en sus comienzos es costoso, ya que se deben emplear muchos esfuerzos en lograr obtener productos de base para reutilizar, pero este proceso se amortiza una vez que se comiencen a obtener resultados.

Técnicas utilizadas para la adaptación de requisitos

Para lograr la adaptación de requisitos una técnica muy conocida es el concepto de herencia asociado a la programación para separar elementos generales de los específicos y de esta forma ambos pueden reutilizar las funcionalidades del padre y posibilita adaptar nuevos requisitos hijos del mismo padre. En la Ingeniería de Requisitos esta técnica se ha utilizado para modelar la herencia entre casos de uso [7].

Otra técnica que se ha utilizado son los patrones debido a que su propia naturaleza propicia la reutilización. Según la definición dada por Christopher Alexander en [27]: "Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la solución a ese problema, de tal manera que esa solución pueda ser usada más de un millón de veces sin hacerlo ni siquiera dos veces de la misma forma. Un ejemplo de estos patrones en la Ingeniería de Requisitos asociados con la adaptación de requisitos es el patrón para casos de uso parametrizables planteados por Cockburn en [28], el cual propone una forma de especificar grupos de requisitos utilizando elementos comunes entre ellos. De esta forma los analistas pueden fomentar adaptación de requisitos mediante la inclusión de nuevos parámetros a la descripción base. Otros ejemplos de patrones son los planteados por Durán en [29] donde se establecen plantillas con comportamientos

estándares para describir casos de uso que se presentan tradicionalmente, como los adicionar, modificar, consultar y eliminar. Mediante la utilización de estos patrones el analista podrá adaptar las descripciones dadas en cada patrón a los nuevos requisitos que desee describir, agilizando así la especificación de requisitos.

En la Figura 5 se muestra a modo de resumen los aspectos vistos en este epígrafe mediante la utilización de un mapa conceptual.

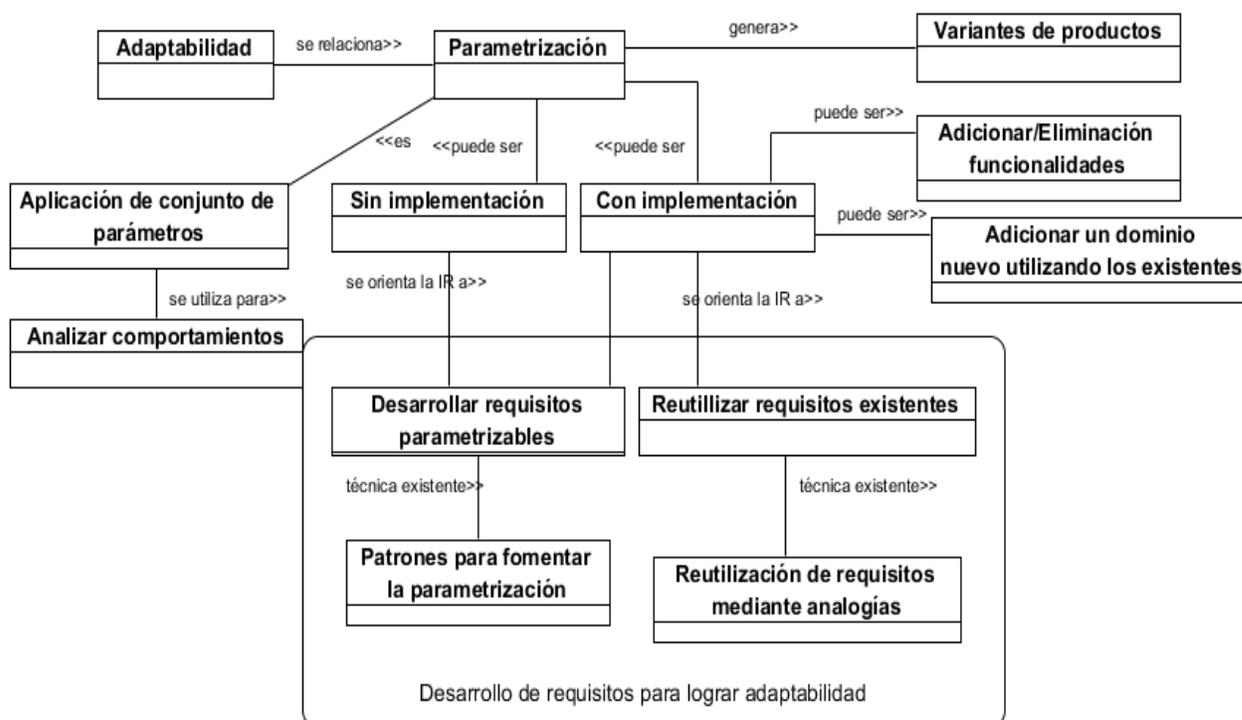


Figura 5 Mapa conceptual para el Desarrollo de Requisitos con vistas a lograr adaptabilidad del sistema

Desarrollo de Requisitos en sistemas modulares

Según Carreón en [4] la modularidad de un sistema está dada por la organización de los procesos de una empresa por módulos, donde estos a pesar de coexistir en un mismo sistema, deben ser capaces de desacoplarse para su instalación en distintos dominios de aplicación. Con el objetivo de facilitar el entendimiento de esta característica es necesario precisar diferentes conceptos como módulo y modularidad.

Según Riba y otros en [30] el término módulo proviene del latín *modulus* que significaba medida o longitud, las medidas estándares que aseguraban unas correctas

proporciones. Este término fue modificado por primera vez por un arquitecto alemán, el cual lo asoció a la construcción de bloques. No es hasta los años 60 cuando se comienza a utilizar este término en la ingeniería y también fue asociado a bloques constructivos, pero esta vez con el objetivo de crear variedad de productos mediante la combinación y el intercambio. Teniendo en cuenta los conceptos dados por Carreón y Riba sobre módulo se define para esta investigación el siguiente concepto.

Módulo: Es una porción o bloque del sistema que es agrupada para lograr un fin específico. En el caso de aplicar este concepto a la Ingeniería de Requisitos quedaría de la siguiente forma: conjunto de requisitos agrupados para realizar las funcionalidades de negocio solicitadas por el cliente. En el caso de los ERP según un estudio comparativo de paquetes de ERP realizado en [31], coincidirían con los dominios funcionales o con paquetes de apoyo a las soluciones de los dominios.

Según Ramos y otros en [32] la modularidad, consiste en el uso de módulos estructuralmente independientes para formar la arquitectura de un producto. Esta división facilita la estandarización de componentes e incrementa la variedad de los productos.

Pressman por su parte en [9] plantea que la modularidad es una propiedad de los sistemas que permite que estos sean intelectualmente manejados. Además provee mecanismos para subdividir un sistema en partes más pequeñas (llamadas módulos), cada una de las cuales debe ser tan independiente como sea posible de la aplicación en sí y de las restantes partes.

Para esta investigación la modularidad se define como: La propiedad con la que cuentan algunos sistemas, que permite la *estructuración del mismo en módulos funcionalmente independientes* para facilitar la estandarización e incrementar la variedad de los productos. Teniendo en cuenta los conceptos antes mencionados se puede llegar a la conclusión de que un sistema puede estar compuesto por módulos y no contener la propiedad de modularidad, por no tener diseñadas sus funcionalidades para trabajar tan independientes como sea posible.

Según Pressman en [9] la independencia funcional entre módulos es la clave para un buen diseño del sistema y ese a su vez es la clave para un software de calidad. Menciona además que esta independencia funcional se evalúa a través de dos criterios: la cohesión y el acoplamiento. La primera es una propiedad interna de cada

módulo, la cual indica la fuerza relativa de las funcionalidades que lo componen para lograr cumplir el objetivo del módulo y el acoplamiento es una propiedad que caracteriza la interdependencia relativa entre módulos.

Para llevar a cabo el Desarrollo de Requisitos para este tipo de sistemas hay que tener en cuenta que las funcionalidades que se determinen deben ser agrupadas en módulos que puedan ser estructuralmente independientes. Esto implica que en este proceso se realice un análisis sobre el agrupamiento de las funcionalidades antes de su implementación.

Técnicas para modularizar productos

Según Ramos y otros en [32] una de las técnicas para modularizar productos son las matrices de análisis de acoplamiento. En estas matrices se plasman las interrelaciones entre las funcionalidades y luego de un análisis del acoplamiento de cada una se realiza un diseño de módulos.

Otra técnica utilizada es el análisis heurístico de las funcionalidades planteado por De La Peña en [33]. Esta técnica se basa en el análisis de las funcionalidades mediante la aplicación de tres heurísticas, las de flujo dominante, donde se analiza el flujo no ramificado de una estructura funcional y agrupa las sub-funciones. El conjunto de sub-funciones identificadas definirán el módulo. La segunda heurística está referida al flujo ramificado y requiere de la identificación de flujos asociados con las cadenas de funciones paralelas. Cada rama de una cadena de funciones paralelas define un módulo potencial. La tercera heurística está de acuerdo con la funciones de conversión y transformación. Una sub-función de conversiones aparece como un flujo A que se convierte en un flujo B. Estas sub-funciones de conversión se convierten en módulos. En la Figura 6 se muestra a modo de resumen los aspectos vistos en este epígrafe mediante la utilización de un mapa conceptual.

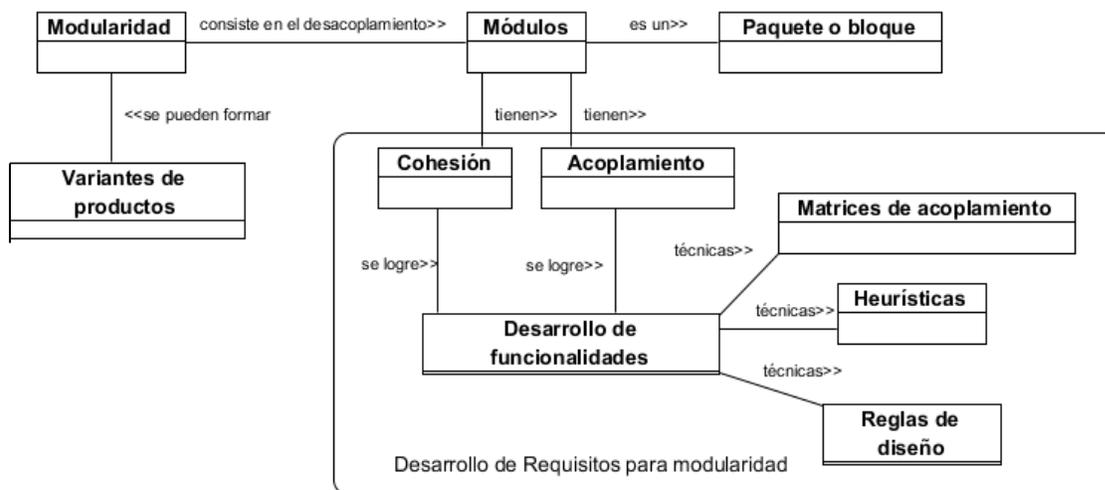


Figura 6 Mapa conceptual para el Desarrollo de Requisitos con vistas a lograr modularidad del sistema

Trabajos relacionados con el Desarrollo de Requisitos para sistemas integrales, modulares y adaptables.

En este epígrafe se realiza una revisión sobre algunos trabajos relacionados con la presente investigación. Se describe brevemente cómo cada uno fomenta alguna de las características de los ERP y se determinan aspectos positivos que pueden tenerse en cuenta para la propuesta de solución y aspectos negativos sobre los cuales se puede incidir. Después de realizar revisiones bibliográficas en búsqueda de procesos de Desarrollo de Requisitos que tengan en cuenta la modularidad, integralidad y adaptabilidad, se encontraron algunos trabajos en los que se fomentan estas características, pero no de forma integrada en un único proceso. En la Tabla 2 se muestran algunos de los trabajos que resultaron de interés para esta investigación. Para discernir en la selección de ellos se tuvo en cuenta los siguientes criterios:

- Actualidad de los trabajos (Preferentemente los últimos 5 años).
- Relación con alguna de las características de los ERP desde la Ingeniería de Requisitos.
- Relevancia ya sea por la trayectoria de los autores en esa área o por el reconocimiento a nivel mundial.

• Tabla 2 Trabajos relacionados

Trabajos relacionados	Modularidad	Adaptabilidad/con implementación	Adaptabilidad/ sin implementación
Desarrollo de Requisitos en SAP/R3			X
El Desarrollo de Requisitos de GESSI (Maestría de catálogo de requisitos para ERP)		X	
Desarrollo de Requisitos en SIREN (Tesis de doctorado en reutilización de requisitos)		X	
Método de diseño para variedad de productos basado en el análisis de su modularidad	X		

Desarrollo de Requisitos en SAP/R3

SAP es un sistema ERP propietario líder en implantaciones según los estudios realizados en [34] y [35]. La compañía que lo realizó es de origen alemán y fue creada por 5 ex-miembros de IBM en Mannheim, Alemania, en 1972.

El Desarrollo de Requisitos para este sistema se realiza de forma automatizada. Para esto sus desarrolladores establecen como principio que los negocios a implementar no son más que vastos mares de datos en estructuras y modelos. A partir de este principio y con un marco de trabajo que según Martín en [36] sustenta los conceptos clave de la gestión de procesos (flujo de trabajo, eventos, tareas, reglas, contextos, roles e interfaz de usuario), modelan el negocio utilizando la Notación para el Modelado de Procesos de Negocio (BPMN) e identifican de cada negocio a implementar sus conceptos que son los que rigen la identificación de los requisitos a implementar.

Una vez identificados los conceptos son almacenados en una estructura llamada Infocubos según Woods en [37] y luego son procesados con el objetivo de hacerlos responder preguntas de acuerdo a los conceptos anteriormente almacenados y conformar los requisitos que debe contener el sistema. Este sistema de trabajo hace que de forma ágil y automatizada se pueda lograr la adaptabilidad del sistema sin implementación.

Aspectos positivos

- Se fomentan las características de un sistema ERP mediante la utilización de la inteligencia de negocios.
- Utiliza herramientas para la generación de aplicaciones similares.
- Por el grado de automatización logran obtener mayor productividad.

Aspectos negativos

- La característica de automatización de los requisitos para el entorno actual de esta investigación no es práctico ni aplicable, pues este marco de trabajo no se puede aplicar por ser propietario y desarrollar uno similar a este implicaría emplear gran cantidad de tiempo en su construcción, lo cual atenta contra la necesidad de resolver los problemas planteados en esta investigación de manera práctica y tempranamente.

Desarrollo de Requisitos de GESSI

El grupo GESSI de España, lleva 10 años en la investigación de la reutilización de requisitos. Mediante la tesis de maestría de Carreón en [4] publicada en el 2008 sobre un catálogo de requisitos funcionales para ERP expone un método para llegar a la formulación de estos patrones, apoyándose en libros de requisitos proporcionados por la empresa consultora CITI de Luxemburgo. A través de la reutilización de requisitos utilizando patrones se fomenta la adaptación de productos mediante la implementación de las partes variables de estos. Las bases de los patrones planteados se sustentan en la determinación a cada requisito de una parte fija y una variable, donde la parte fija podrá reutilizarse siempre y en la variable se podrá incluir las nuevas adaptaciones. Además utilizando esta técnica y los libros de requisitos proporcionados por la empresa consultora antes mencionada, establece un conjunto de requisitos clásicos de ERP por tipos de dominios.

Aspectos positivos

- La técnica para describir los requisitos reutilizables es práctica y aplicable a cualquier tipo de sistema, no solo a los ERP.
- El catálogo de requisitos resultante fue automatizado.
- El método cuenta como entrada un libro de requisitos certificados.
- Cuenta con una definición de módulos tipos para un ERP.

Aspectos negativos

- La plantilla definida para describir los requisitos cuenta con elementos que no son imprescindibles y que pueden demorar la actividad de especificación, tal es el caso de los campos, métricas, tipo de métricas. Además no incluye aspectos estándares de las plantillas de especificación de requisitos como los flujos alternos, escenarios de error.
- No especifica cómo identificar las coincidencias del requisito nuevo con el almacenado.
- Solo se especifican requisitos de alto nivel.

Desarrollo de Requisitos en SIREN (Reutilización sencilla de requisitos de software)

El método SIREN es desarrollado por el grupo de investigación en Ingeniería de software SIREN de la Universidad de Murcia en España. El método se aprueba con la tesis doctoral cita en [26] en el 2009. Este propone una forma de realización de la Ingeniería de Requisitos con vistas a la reutilización de software en líneas de productos basado en 8 actividades. Está basado en la reutilización de requisitos de alto nivel y tiene en cuenta también la parametrización de requisitos. Su reutilización se centra básicamente en la reutilización de las especificaciones de requisitos durante la construcción de nuevos productos. Las actividades que define este método son las siguientes: Creación del catálogo de requisitos reutilizables, Gestión del proceso de requisito, Reutilización de requisitos, Extracción de requisitos del proyecto actual, Análisis y negociación de requisitos, Documentación de requisitos, Validación de requisitos, Mejora del repositorio.

Aspectos positivos.

- Se registran las lecciones aprendidas durante el desarrollo de cada producto.
- Cuenta con guías para la aplicación del método basadas en estándares de requisitos de la IEEE.
- Es un proceso sistemático.
- Incorpora a las actividades de la Ingeniería de Requisitos actividades de reutilización.

Aspectos negativos.

- No fomenta todas las características de un ERP, solo la adaptabilidad y parametrización de un producto con reutilización.
- No presenta patrones léxicos de requisitos lo cual es indispensable a la hora de realizar comparaciones entre los requisitos utilizando este método.
- No establece cómo realizar las búsquedas en repositorios de documentación.
- Ata el método a la utilización de plantillas, cuando la utilización de aspectos gráficos en los procesos de análisis es muy recomendado para ganar en comprensión.

Método de diseño para variedad de productos basado en el análisis de su modularidad.

Es un método desarrollado por un grupo de Doctores en Ingeniería Mecánica de la Universidad Nacional Autónoma de México y publicado en el 2008 cito en [32]. Este método resultó de interés para esta investigación, pues realiza análisis sobre requisitos funcionales para la creación de módulos. Para esto utiliza la técnica de análisis de matrices.

En el método se proponen los siguientes pasos:

- Estudio comparativo (benchmarking), para ubicar al producto con respecto a competidores.
- El uso de matrices de acoplamiento.
- Cuantificar los índices de variedad generacional
- Cuantificar los índices de acoplamiento para la arquitectura del producto actual.
- Jerarquización y uso de la matriz de diseño de la estructura para especificar posibles módulos.
- Realización de un ciclo iterativo entre la reducción de los índices y opciones de arquitectura
- Realización de una evaluación de los índices para la nueva arquitectura [32].

Aspectos positivos

- Es un método práctico de aplicar.
- Tiene en cuenta la traza entre requisitos del producto y los componentes del sistema mediante la utilización de matrices.

Aspectos negativos

- No tiene en cuenta el desacoplamiento de requisitos a un nivel más alto como el negocio.

Luego de estudiar cada trabajo, se puede concluir que todos ofrecen soluciones parciales a la necesidad de sistemas modulares, integrales y adaptables. Como en esta investigación se requiere fomentar dos de las características antes mencionadas, se hace necesario el diseño de un nuevo proceso que las promueva.

Conclusiones parciales

Luego de haber realizado un estado del arte sobre la Ingeniería de Requisitos para sistemas modulares, integrales y adaptables, se puede concluir lo siguiente:

- Es necesario para el Desarrollo de Requisitos de estos sistemas realizar acciones desde etapas tempranas enfocadas al cumplimiento de cada una de sus características.
- A pesar de todos los aspectos positivos de los trabajos analizados, hasta el momento ninguno satisface de forma íntegra el Desarrollo de Requisitos orientado a lograr modularidad y adaptabilidad.
- La reutilización de requisitos presume una ventaja en el desarrollo de software a pesar de que el costo inicial para comenzar el desarrollo es alto.

CAPÍTULO 2 DESARROLLO DE REQUISITOS PARA SISTEMAS INTEGRALES, MODULARES Y ADAPTABLES

Introducción

En el presente capítulo se describe el proceso para desarrollar requisitos en sistemas de gestión integrales, modulares y adaptables. Primeramente se presentan algunos aspectos importantes a tener en cuenta para el entendimiento de la propuesta, luego se realiza la modelación de las actividades así como los artefactos a generar y los roles que deben participar en cada caso. En esta solución se hace referencia a las buenas prácticas de los trabajos relacionados que han formado parte del estudio del estado del arte.

Consideraciones generales de la propuesta

La solución que se propone para llevar a cabo el Desarrollo de Requisitos integra los siguientes elementos:

- El proceso tradicional de Desarrollo de Requisitos
- Análisis de requisitos para lograr adaptabilidad del sistema
 - Actividades, roles y artefactos
- Análisis de requisitos para lograr modularidad del sistema
 - Actividades , roles y artefactos

Proceso de Ingeniería de Requisitos seleccionado

Necesidad: Definición del proceso base de Ingeniería de Requisitos que soportará la propuesta.

Solución: Después de haber realizado la comparación en el capítulo anterior sobre las corrientes de la Ingeniería de Requisitos por autores de renombre en la literatura se define como proceso base para sustentar esta propuesta las actividades que resultaron comunes y se le añade una actividad de almacenamiento con el objetivo de definir la estructura de almacenamiento particular para este tipo de sistemas integrados. El resultado final se muestra en la Figura 7.



Figura 7 Desarrollo base de Ingeniería de Requisitos

Para apoyar la etapa de elicitación de requisitos se propone la utilización de requisitos tradicionales en un software de gestión. Después de un análisis de más de 200 requisitos del producto se pueden proponer requisitos tradicionales de un software de gestión, así como su comportamiento base estándar, los que se listan a continuación:

- Adicionar/Crear <<concepto en singular>>
- Modificar <<concepto en singular>>
- Eliminar <<concepto en singular>>
- Listar <<concepto en plural>>
- Ordenar <<concepto en plural>>
- Consultar <<concepto en singular>>
- Buscar <<concepto en singular>>
- Importar <<concepto en plural>>
- Exportar <<concepto en plural>>
- Imprimir <<concepto>>

Para la descripción de estos requisitos, se han definido patrones de comportamientos bases, de forma tal que el analista solo tenga que ajustar las especificaciones al nuevo contexto. Esto trae como ventajas:

- El analista puede describir los requisitos de manera más ágil, pues cuenta con una serie de pasos ya establecidos.
- El analista evita omitir algunos elementos repetitivos, por ejemplo los tipos de flujos alternos en cada caso.

En el Anexo 1 se muestra un ejemplo del patrón de Adicionar/Crear <<concepto en singular>>

Análisis de requisitos para lograr adaptabilidad del sistema con implementación

Necesidad: Reutilizar de forma total o parcial los requisitos definidos para un producto en la construcción de otros.

Solución: Se propone utilizar una técnica basada en el razonamiento humano, mediante la cual se utiliza la experiencia acumulada en la realización de nuevas soluciones, mediante esta se realizan las siguientes actividades :

- Buscar elementos similares
- Comparar y Adaptar
- Almacenar

Con el objetivo de orientarlo a reutilizar requisitos el proceso quedaría como se muestra en la Figura 8

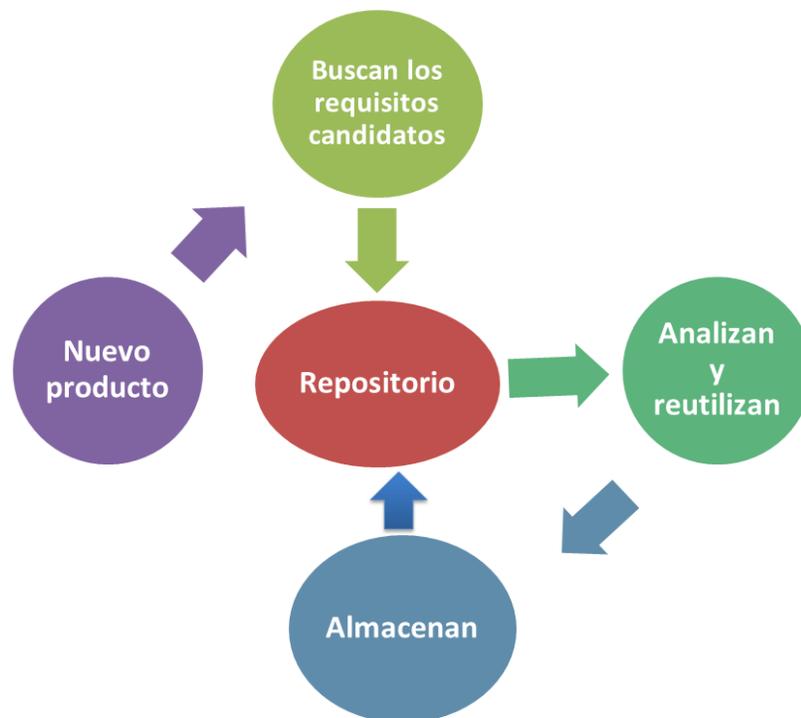


Figura 8 Proceso de reutilización orientado a requisitos

- Nuevo producto: El cliente solicita la realización de un nuevo producto.
- Buscar requisitos candidatos: El objetivo de esta actividad es localizar en el repositorio los requisitos que puedan ser utilizados en la nueva solución.

Normalización de requisitos

Para realizar la búsqueda los requisitos deben estar normalizados y para esto se propone el siguiente patrón:

<<Acción>><<Concepto/frase nominal>> <<Espacios de variación>>

Parte fija o común Parte variable

- Acción: Es un verbo, por ejemplo Gestionar, Realizar
- Concepto/frase nominal: Se refiere al concepto de negocio o frase nominal sobre el cual se va a realizar la acción, por ejemplo cierres contables, liquidación de derechos
- Espacios de variación: Se refiere a parámetros que hagan variar el requisito, por ejemplo liquidar anticipos de derecho de cobro o de pago anticipado.

En el Anexo 2 se muestra una plantilla ajustada para describir estos requisitos.

Una vez estandarizados los requisitos se procede a la búsqueda en el repositorio a través de la identificación de similitudes por analogías, la comparación se realizará utilizando la modelación jerárquica de los requisitos del cliente.

En este punto se pueden presentar varios casos:

Tabla 3 Acciones a realizar para recuperar requisitos

<p>Caso 1</p> 	<p>Existen requisitos en el nuevo producto que se asemejan a los requisitos de otros productos, total o parcialmente.</p>	<p>Se recuperan los similares</p>
--	---	-----------------------------------

Caso 2 	No existen requisitos semejantes entre el nuevo producto y los existentes	No se recupera nada
Caso 3 	Los requisitos del nuevo producto coinciden exactamente con los de otro producto.	Se recupera todo

Para apoyar la selección de los requisitos del producto se realizará una matriz de requisitos del cliente contra requisitos del producto. Una vez encontradas las similitudes entre los requisitos del cliente, se procede a localizar los requisitos del producto que coinciden con los requisitos del cliente reutilizados. Se debe analizar también la matriz de dependencias entre requisitos del cliente para recuperar además aquellos requisitos que aunque no están directamente asociados al requisito del cliente reutilizado, es necesario que sea reutilizado para su correcto funcionamiento.

Tabla 4 Matriz de dependencias entre requisitos del cliente y requisitos del productos

PRODUCTOS		Requisito del cliente 1	Requisito del cliente 2	Requisito del cliente 3
		Producto<<versión>>	Adicionar <<concepto>>	1
	Modificar <<concepto>>	0	1	1

Tabla 5 Matriz de dependencias entre requisitos del cliente

PRODUCTOS		Requisito del cliente 1	Requisito del cliente 2	Requisito del cliente 3
	Producto<<versión>>	Requisito del cliente 1	0	1
	Requisito del cliente 2	0	0	1

- Analizar y reutilizar requisitos: El objetivo de esta actividad es a partir de un análisis que se realice de los requisitos realizar las modificaciones pertinentes (en caso de ser necesario). Para realizar este análisis se utilizará una matriz de dependencias entre requisitos del cliente y el artefacto de análisis de reutilización que se muestra en la Figura 9 .Durante el análisis se pueden presentar los siguientes casos:

Tabla 6 Acciones por casos de reutilización

<p>Caso 1</p> 	<p>Existen requisitos en el nuevo producto que se asemejan a los requisitos de otros productos, total o parcialmente.</p>	<p>Se reutilizan los requisitos de los productos relacionados, si se reutilizan totalmente solamente se hace referencia al producto, si se reutilizan parcialmente o por adición de elementos se realizan las modificaciones pertinentes.</p>
<p>Caso 2</p>	<p>No existen requisitos semejantes entre el nuevo producto y los</p>	<p>Se desarrolla la Ingeniería de Requisitos</p>

	existentes	de la forma tradicional.
Caso 3 	Los requisitos del nuevo producto coinciden exactamente con los de otro producto.	Se referencian todos los requisitos

Requisitos	Referencia	Requisitos nuevos	Conceptos nuevos	Restricciones nuevas	Otra modificación	Observaciones	Nivel de reutilización

Figura 9 Artefacto para reutilizar requisitos

- Almacenar: El objetivo de esta actividad es una vez estables los requisitos proceder a almacenarlos en el repositorio. Antes de almacenarlo se revisa y se refina la información. De esta manera se contribuye a mejorar el mismo ya que se nutre de mayor cantidad de requisitos para comparar cuando se vaya a desarrollar el nuevo producto. La estructura de almacenamiento que se propone se muestra en la Figura 10. Esta estructura tiene como elemento particular el almacenamiento de las integraciones de los productos.

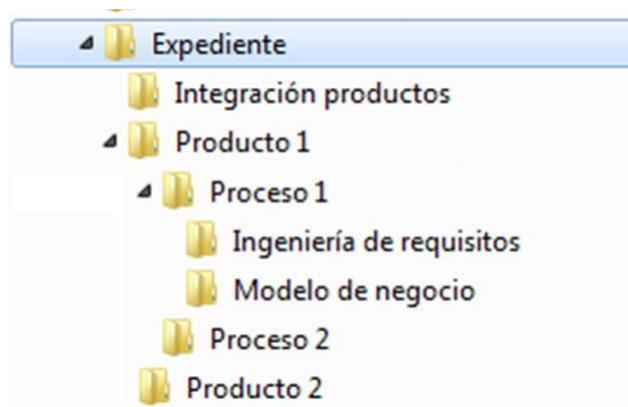


Figura 10 Estructura del repositorio

Análisis de requisitos para lograr adaptabilidad del sistema sin implementación

Necesidad: Necesidad de que el sistema se ajuste a la mayor cantidad de dominios posibles.

Solución: La adaptabilidad del sistema se logra mediante las parametrizaciones que se le realicen a este, ya que esto le aportará mayor capacidad de adaptación a distintos dominios de aplicación. Para la identificación de las parametrizaciones se propone realizar un análisis de los requisitos utilizando el patrón de normalización anteriormente descrito. El análisis sobre el patrón se debe realizar como se muestra en la Figura 11.

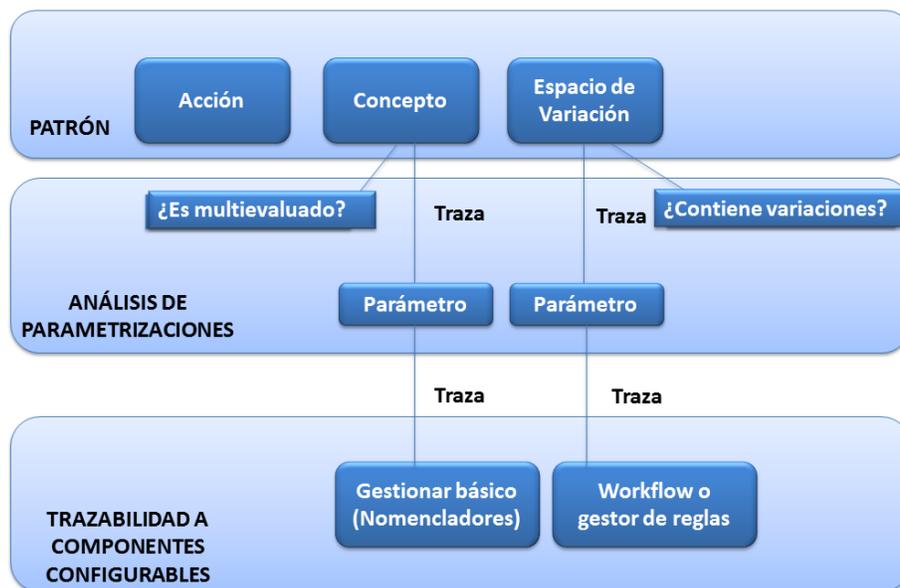


Figura 11 Análisis de parametrizaciones

Se convertirán en requisitos parametrizables aquellos requisitos que cumplan alguna de las siguientes condiciones:

- El concepto principal es multievaluado, por tanto se convierte en un parámetro que tomará diversos valores dependiendo del dominio de implantación.
- El requisito contiene espacio de variación, por tanto cada uno se convertirá en parámetros del requisito base. Mientras más espacios de variación contenga un requisito mayor probabilidad de adaptación tendrá.

A partir de los requisitos parametrizables identificados se convertirán en configuraciones generales o particulares del sistema. En las configuraciones generales se incluyen aquellos requisitos parametrizables que sean comunes a varios módulos y en las configuraciones particulares aquellos requisitos parametrizables que sean comunes a varias funcionalidades de los procesos que integran un módulo.

Para la descripción de las parametrizaciones

- Cuando el concepto del requisito es multievaluado: Se utilizarán las plantillas de los requisitos tradicionales donde se especifican los comportamientos recurrentes
- Cuando existen espacios de variación se utilizarán las plantillas parametrizables.

Análisis de requisitos para lograr modularidad del sistema

Necesidad: Necesidad de identificar la modularidad desde etapas tempranas, para evitar que se formen módulos erróneos, afectando así la cohesión de las funcionalidades que lo componen y evitar además que no se pueda desacoplar para formar variantes de productos.

Solución: Para apoyar la identificación de la modularidad se propone la utilización de una taxonomía donde se homologan los niveles de procesos con las parte de un sistema. De esta forma mediante la determinación del alcance de la solución se contribuye a lograr cohesión entre las funcionalidades a la hora de agruparlas en módulos. Se propone además añadir a este modelo la representación de los parámetros que hacen variar a un requisito de cliente. Es válido señalar que el proceso de nivel 0 coincidirá para el caso de los sistemas ERP, con alguna de las áreas funcionales de las empresas. En la Figura 12 se muestra un ejemplo de cómo quedaría esta taxonomía y su homologación.

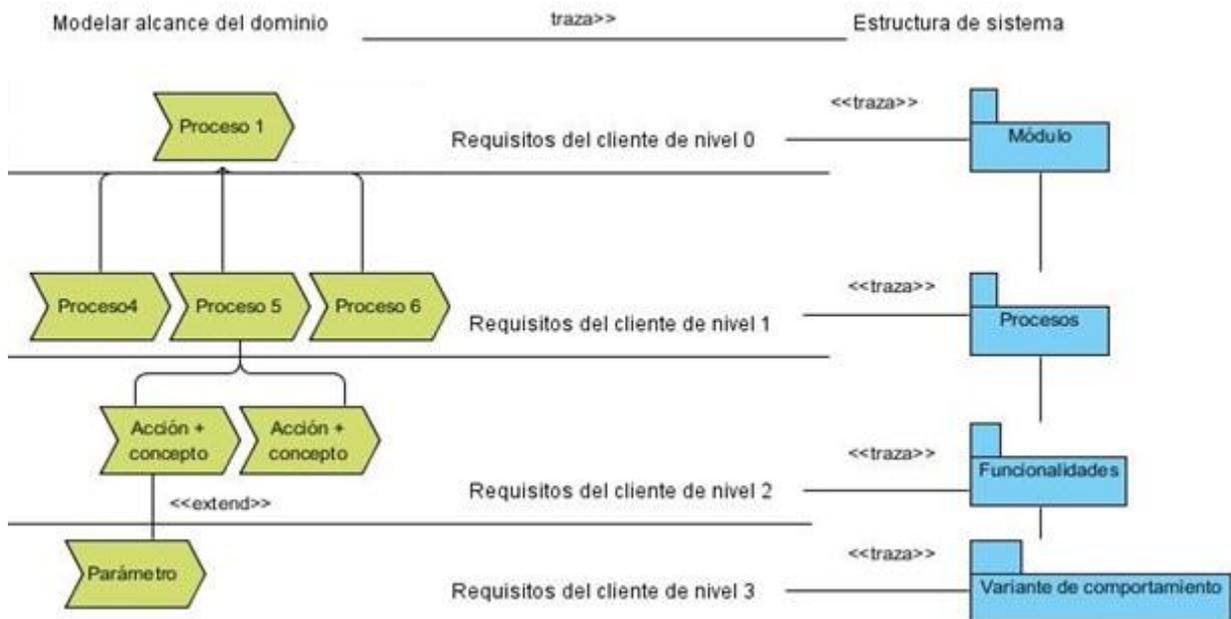


Figura 12 Homologación de componentes del sistema con una organización

Una vez que se tiene identificado el alcance de funcionalidades que debe contener cada módulo se deben analizar las dependencias funcionales que existen entre ellos, para esto es necesario obtener una visión integrada de los requisitos del sistema que se construya, por tanto se propone la realización de las siguientes vistas:

- Vista integrada de conceptos de negocio.
- Vista integrada de requisitos del cliente.

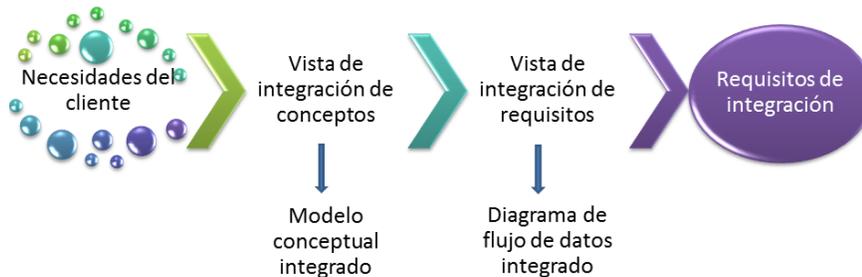


Figura 13 Vistas de integración

- Vista integrada de conceptos de negocio: Esta vista tiene como objetivo identificar puntos de integración a nivel conceptual entre dominios. Estos puntos de integración constituyen una entrada para la vista integrada de requisitos del cliente.

- Artefacto: Modelo conceptual integrado
- Descripción: Representa los conceptos de negocio sus interrelaciones. Para representar integración con otro dominio se utilizarán paquetes de agrupamiento.
- Ejemplo:

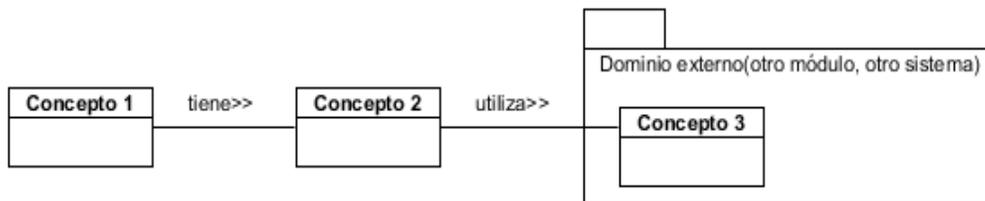


Figura 14 Modelo conceptual integrado

- Vista integrada de requisitos del cliente: Esta vista identifica las interrelaciones entre los requisitos del cliente y los conceptos de dominio identificados anteriormente. Esto es muy importante en esta etapa, pues de cada interrelación surgen nuevos requisitos (en lo adelante requisitos de integración). Estos requisitos luego se deben entregar a los arquitectos para que garanticen una comunicación entre módulos y/o sistemas externos.
 - Artefacto: Diagrama de flujo de datos integrado
 - Descripción: En este mapa se representarán los requisitos del cliente con los conceptos que los interrelacionan. Cada una de estas interrelaciones de datos generarán requisitos de integración entre ambos requisitos.

- Ejemplo:

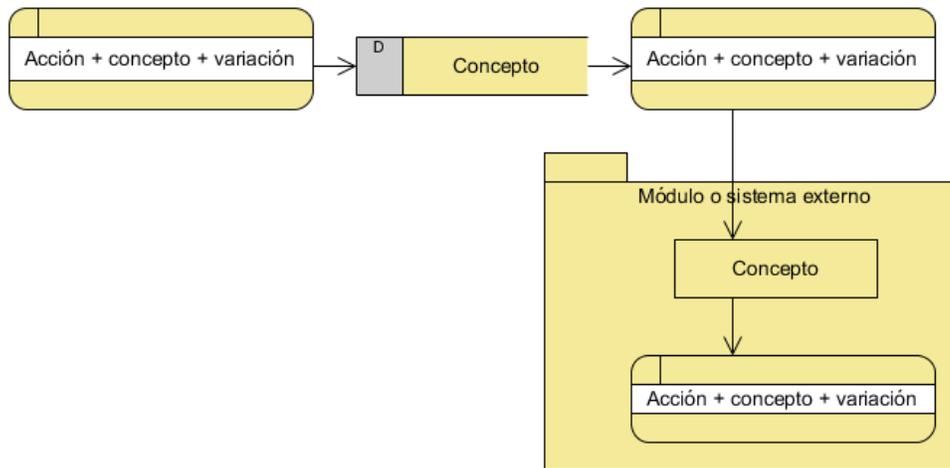


Figura 15 Diagrama de flujo de datos integrado

- Artefacto: Descripción de requisitos de integración
- Descripción: Mediante este artefacto se describirán los requisitos que surjan de las integraciones entre requisitos del cliente.
- Ejemplo: Se utilizará la misma plantilla para describir el resto de los requisitos, pero en este caso se especificarán solamente las precondiciones, atributos, validaciones y post-condiciones.

Una vez identificadas las integraciones entre módulos se propone la realización de los siguientes pasos.

- Eliminación de todas las integraciones directas de requisitos entre módulos.
- Formulación de requisitos de integración entre los módulos acoplados.
- Identificación final de módulos incluyendo el módulo que manejará únicamente las integraciones.

En la Figura 16 se muestra un diagrama de cómo debe realizarse el análisis y su trazabilidad luego al sistema.

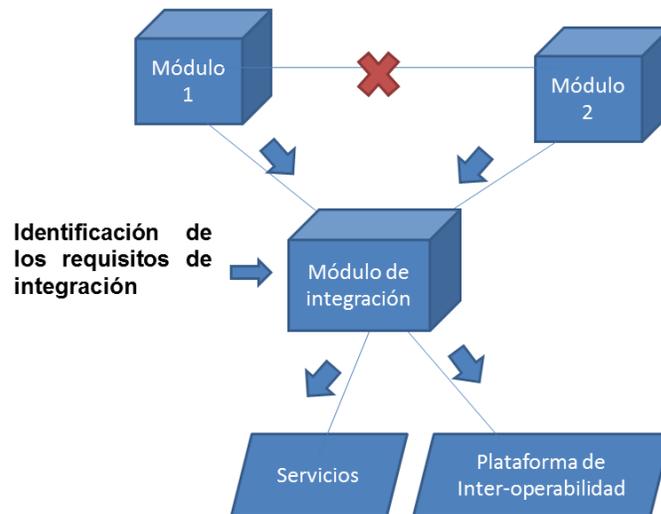
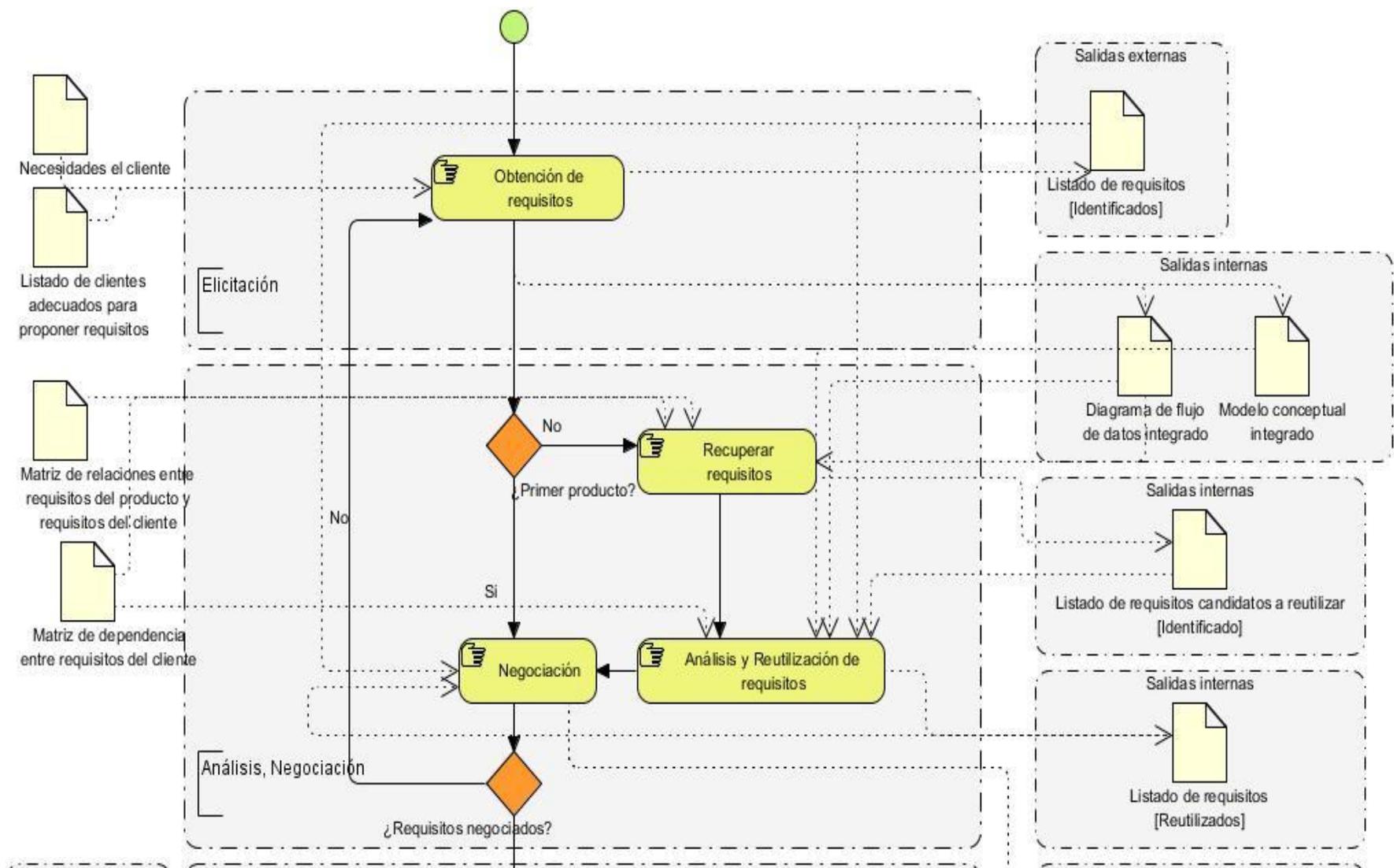


Figura 16 Regla de diseño establecida para lograr modularidad

Los requisitos de integración que se proponen relacionarán tanto la información que brindan a otro módulo o sistema como la que reciben y recogerán también las reglas mediante las cuales se utiliza información compartida. Luego estos requisitos deben ser implementados en el sistema como una capa de servicios entre módulos o una plataforma de inter-operabilidad.

Modelado y descripción del proceso de Desarrollo de Requisitos para sistemas modulares, integrales y adaptables

Diagrama del proceso de Desarrollo de Requisitos integrado



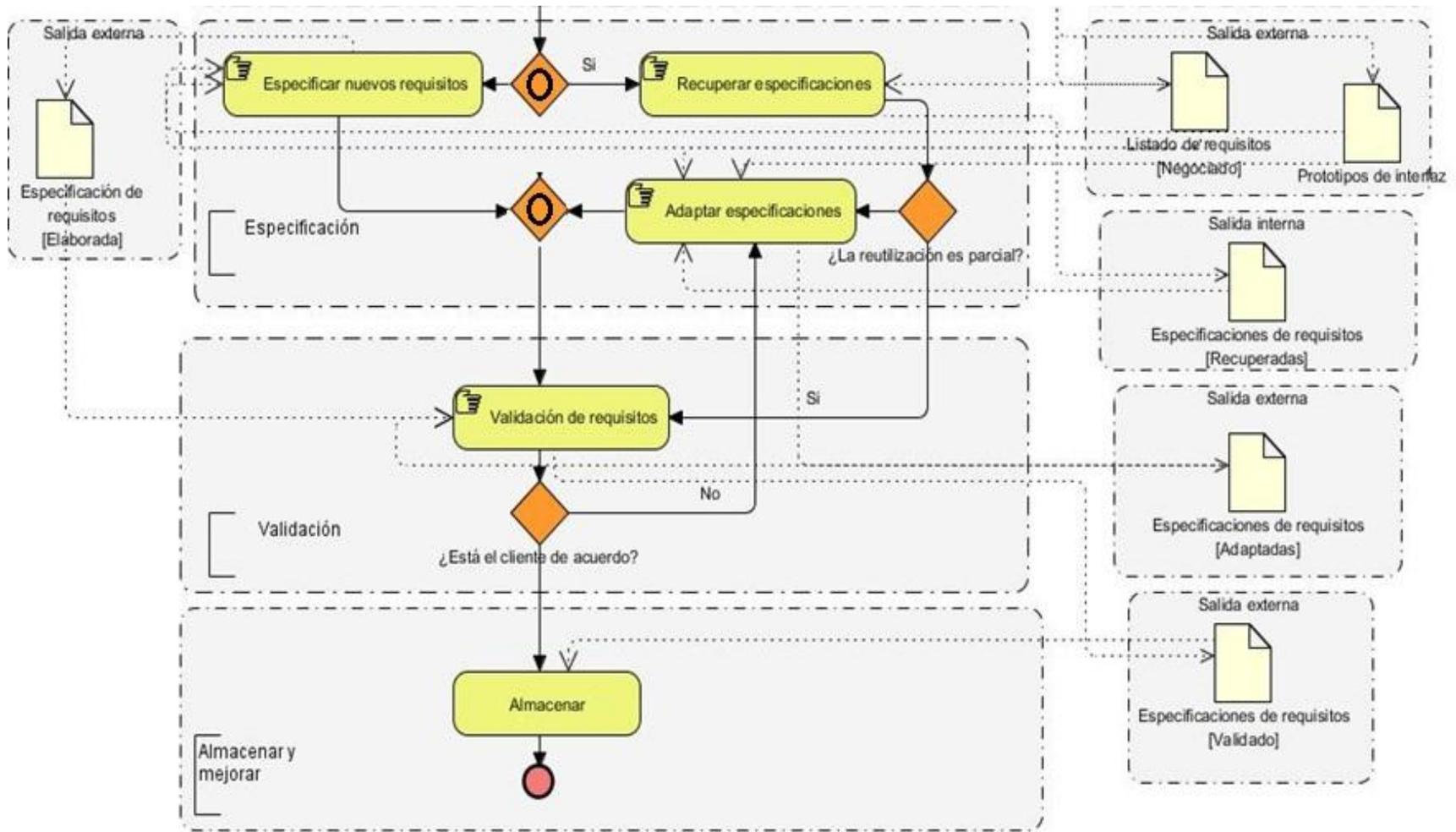


Figura 17 Diagrama del proceso de Desarrollo de Requisitos propuesto

Descripción de actividades del proceso de Desarrollo de Requisitos [Cliente, Producto]

Nombre	Obtención de requisitos
Objetivo	Identificar los requisitos que el cliente desea que contenga el futuro sistema.
Entradas	Necesidades del cliente Listado de clientes adecuados para proponer requisitos
Salidas	Externas <ul style="list-style-type: none">○ Listado de requisitos [Identificados] Internas: <ul style="list-style-type: none">○ Modelo conceptual integrado○ Diagrama de flujo de datos integrado
Responsable	Analista
Involucrados	Cliente, Analista
Descripción	<p>El analista para identificar los requisitos del cliente debe realizar las siguientes tareas:</p> <ul style="list-style-type: none">○ Normalizar los requisitos○ Modelar los requisitos○ Identificar parametrizaciones○ Identificar integraciones <p>Para el caso de identificar requisitos del producto debe verificar si hay reutilización en los requisitos del cliente</p> <p>Vía 1: Hay reutilización: El analista debe pasar a la actividad recuperar requisitos candidatos.</p>

	<p>Vía 2: No hay reutilización: Debe identificar los requisitos utilizando el patrón de normalización.</p> <p>En todos los casos se deben refinar los artefactos realizados en la identificación de requisitos del cliente.</p>
Técnicas	<p>Tormenta de ideas</p> <p>Revisión de documentos</p> <p>Utilización de los requisitos tradicionales en software de gestión</p> <p>Utilización del patrón de normalización</p>
Herramientas	Visual Paradigm, Hoja de cálculo

Nombre	Recuperar requisitos
Objetivo	Identificar desde etapas tempranas de la construcción del software los requisitos del cliente candidatos a reutilizar.
Entradas	<p>Requisitos del cliente</p> <p>Diagrama de flujo de datos integrado</p> <p>Requisitos del producto</p> <p>Matriz de relaciones entre requisitos del producto y requisitos del cliente</p> <p>Matriz de dependencia entre requisitos del cliente.</p>
Salidas	<p>Internas</p> <ul style="list-style-type: none"> ○ Listado de requisitos candidatos a reutilizar
Responsable	Administrador del repositorio
Involucrados	Analista, Administrador del repositorio

Descripción	<p>Para requisitos del cliente</p> <p>El responsable busca en el repositorio los diagramas de flujo de datos integrado y realiza analogías con el identificado para el nuevo producto. A partir de esto identifica cuáles de los existentes en el Repositorio son similares y los recupera.</p> <p>Para requisitos del producto</p> <p>El responsable a partir de los requisitos del cliente que se pudieron reutilizar, localiza las matrices de interrelación de requisitos existentes en el repositorio de los productos que resultaron similares desde la identificación de los requisitos del cliente.</p> <p>Selecciona como candidatos todos los requisitos del producto que corresponden con los requisitos del cliente reutilizados. Se debe analizar además la matriz de dependencias entre los requisitos del cliente, pues al reutilizar un requisito, puede ser necesario reutilizar algún otro que dependa de él.</p>
Técnicas	Analogías entre requisitos
Herramientas	<p>Se utilizará la funcionalidad Visual Diff de Visual Paradigm para realizar las comparaciones.</p> <ul style="list-style-type: none"> ○ Interpretación de las analogías resultantes: Los que están en estado Modificado son los requisitos iguales en ambos productos, los que están en estado Nuevo son los requisitos que están en uno de los productos almacenados y no en el producto nuevo, estos pueden ser reutilizados para proponerlo al cliente. Los que tienen el estado Eliminado son los requisitos propios del producto nuevo.

Nombre	Análisis y Reutilización de requisitos
Objetivo	Utilizar requisitos de otros productos que puedan ser aplicables al nuevo producto y que hayan sido identificados o no previamente.
Entradas	Listado de requisitos candidatos a reutilizar Listado de requisitos [Identificado Diagrama de flujo de datos integrado Modelo conceptual integrado Matriz de dependencia entre requisitos del cliente
Salidas	Listado de requisitos [Reutilizados]
Responsable	Analista
Involucrados	Analista, Administrador del repositorio.
Descripción	Se analizará el listado de requisitos candidatos a reutilizar y se seleccionarán los que de acuerdo al negocio son efectivamente reutilizables. Pueden utilizarse como apoyo los artefactos realizados en la actividad de Obtención de requisitos. Luego se analizará el nivel de reutilización que pueden tener, indicando los aspectos a adaptar, a incluir o a eliminar.
Técnicas	Análisis de documentos
Herramientas	Visual Paradigm, Hoja de cálculo

Nombre	Negociar requisitos
Objetivo	Llegar a un acuerdo entre clientes y analistas sobre los requisitos del cliente que contendrá el sistema.

Entradas	Listado de requisitos [reutilizados y/o no reutilizados]
Salidas	Salidas externas <ul style="list-style-type: none"> ○ Listado de requisitos [Negociado] ○ Prototipos de interfaz
Responsable	Analista
Involucrados	Analista, cliente
Descripción	El analista le presenta al cliente los requisitos identificados y refina los requisitos con el cliente. En caso de que fruto de la reutilización hayan surgido requisitos nuevos posibles a reutilizar y que no se hubiesen identificado, el analista debe llegar a un acuerdo con el cliente sobre la inclusión o no del requisito. De no estar de acuerdo con algún requisito se pasa entonces a la actividad de Obtención de requisitos del cliente y conceptos del negocio.
Técnicas	Se realiza mediante un taller.
Herramientas	Hoja de cálculo, Axure

Nombre	Especificar nuevos requisitos
Objetivo	Describir los requisitos identificados.
Entradas	Listado de requisitos [Negociado]. Prototipos de interfaz
Salidas	Salida externa <ul style="list-style-type: none"> ○ Especificación de requisitos [Elaborada]

Responsable	Analista
Involucrados	Analista
Descripción	<p>El analista describe los requisitos del cliente utilizando la plantilla definida para esto.</p> <p>Para requisitos del producto</p> <p>El analista describe los requisitos del producto con mayor nivel de detalle, utilizando la plantilla definida para esto.</p>
Técnicas	<p>Se utilizarán las plantillas donde se describen patrones de comportamiento de los requisitos recurrentes en un software de gestión.</p> <p>Se utilizarán las plantillas para describir las parametrizaciones.</p>
Herramientas	Hoja de texto

Nombre	Recuperar especificaciones de requisitos
Objetivo	Obtener las especificaciones de requisitos que existan en el repositorio para poder reutilizarlas.
Entradas	Listado de requisitos [Negociado].
Salidas	<p>Salida interna</p> <ul style="list-style-type: none"> ○ Especificaciones de requisitos [Recuperadas]
Responsable	Administrador del repositorio
Involucrados	Administrador del repositorio, Analista
Descripción	Utilizando la referencia de cada requisito se localiza en el repositorio las especificaciones a reutilizar.

Técnicas	N/A
Herramientas	N/A

Nombre	Adaptar especificaciones de requisitos
Objetivo	Ajustar las especificaciones de requisitos que se recuperaron para minimizar el esfuerzo y tiempo a la hora de describir.
Entradas	Listado de requisitos [Negociado]. Prototipos de interfaz Especificaciones de requisitos [Recuperadas]
Salidas	Salida externa <ul style="list-style-type: none"> ○ Especificación de requisitos [Adaptadas].
Responsable	Analista
Involucrados	Administrador del repositorio, Analista
Descripción	Una vez obtenidas las especificaciones pueden ser reutilizadas de forma total, cuando coincide exactamente toda la descripción con lo que desea el cliente o se puede reutilizar de forma parcial, cuando solo hay que realizarle algunas modificaciones. En caso de reutilización total solamente se debe referenciar la especificación de requisitos y en caso de ser parcial se debe copiar y ajustar la especificación.
Técnicas	N/A
Herramientas	Hoja de texto

Nombre	Validar especificaciones de requisitos
Objetivo	Obtener una aprobación de los requisitos por parte del cliente.
Entradas	Especificación de requisitos [Elaborada]. Especificación de requisitos [Adaptadas].
Salidas	Salidas externas <ul style="list-style-type: none"> ○ Especificación de requisitos [Validado].
Responsable	Analista
Involucrados	Cliente, Analista
Descripción	Se debe realizar una revisión formal por parte de los clientes donde se revise cada descripción de requisitos. En caso de no aprobarse algún requisito vuelve a la actividad de Adaptar especificaciones. Se deben revisar además los prototipos de interfaz.
Técnicas	Taller Análisis de documentos
Herramientas	Hoja de texto

Nombre	Almacenar
Objetivo	Añadir al repositorio los requisitos [Validado]
Entradas	Especificación de requisitos [Validado].
Salidas	N/A
Responsable	Administrador del repositorio

Involucrados	Administrador del repositorio
Descripción	Se almacenará en el repositorio las especificaciones de requisitos del cliente realizadas según la estructura definida para esto.
Técnicas	N/A
Herramientas	Control de versiones

Conclusiones parciales

Al concluir este capítulo se puede arribar a las siguientes conclusiones:

- Se obtiene un proceso de Desarrollo de Requisitos ajustado a las condiciones de los sistemas modulares, integrales y adaptables.
- Permite reutilizar requisitos definidos para un producto durante la construcción de otros.
- Contribuye a que los analistas puedan determinar requisitos que hasta el momento estaban ocultos, como los requisitos de integración.
- Contribuye a lograr mayor adaptabilidad del sistema a distintos dominios.
- Contribuye a lograr uniformidad en los requisitos desarrollados por diversas personas.

CAPÍTULO 3 VALIDACIÓN DE LA PROPUESTA

Introducción

En este capítulo se evalúa la aplicación del proceso desarrollado en el capítulo 2, utilizando como experiencia los productos del proyecto ERP-Cuba. La propuesta presentada se aplicó posterior a la realización de las pruebas de los productos, pues los artefactos y las actividades se fueron elaborando en la medida que se incurrieron en errores durante el desarrollo de los mismos. Una vez certificados los productos se realizó una re-ingeniería de cada uno de manera tal que sirviera para:

- Dejar formalizados correctamente los productos para nuevos desarrollos partiendo de los anteriores elaborados.
- Evaluar el grado de aceptación de la propuesta a partir del análisis de los errores en los que se incurrieron y que podían haber sido evitados.

La aplicación de esta propuesta sirve como guía para su utilización en otros proyectos con características similares a los seleccionados.

Población y selección de la muestra

El proyecto ERP-Cuba, cuenta con 14 soluciones, las cuales se encuentran en distintos estados y con características similares. De las 14 soluciones se seleccionaron de forma intencionada las 10 que implementaban áreas de procesos de las empresas, pues el resto de las soluciones lo constituyen paquetes técnicos de apoyo a la gestión de los procesos. Además para minimizar los errores en el análisis de los requisitos estos debían contar con un alto grado de estabilidad. Para el caso de la adaptabilidad con implementación se seleccionó el producto Mantenimiento Vehicular para la República Bolivariana de Venezuela por ser un producto que debía tomar como base algunos de las 10 soluciones previamente elaboradas. En la Tabla 7 se muestran los proyectos seleccionados.

Tabla 7 Productos del ERP seleccionados

Productos	Solución modular, integral y adaptable	Fase concluida
Contabilidad	Sí	Certificación
Costos y Procesos	Sí	Certificación
Caja	Sí	Certificación
Banco	Sí	Certificación
Cobros y Pagos	Sí	Certificación
Capital Humano	Sí	Certificación
Estructura y Composición	Sí	Certificación
Inventario	Sí	Certificación
Facturación	Sí	Certificación
Activo Fijo Tangible	Sí	Pruebas internas

Establecimiento de las variables

Variable independiente: Proceso de Desarrollo de Requisitos

Variables dependientes: Grado de modularidad y adaptabilidad de los productos.

En la Tabla 8 se muestra el desglose de las variables y sus respectivas mediciones.

Tabla 8 Desglose de variables dependientes y sus respectivas mediciones

Variables	Indicador	Medición	Fórmula	Interpretación
Proceso de Desarrollo de Requisitos	Aplicabilidad	¿El proceso incluye análisis para lograr modularidad y adaptabilidad de los productos?	No aplica	Sí o No
		¿El proceso incluye actividades, artefactos y roles para apoyar la modularidad y adaptabilidad de los productos?	No aplica	Sí o No
Modularidad	Grado de cohesión	Determinar cuán cohesionadas se encuentran los requisitos de un módulo para cumplir el objetivo del dominio por el cual fue considerado módulo.	$G_C = \text{Cantidad de funcionalidades para cumplir el objetivo módulo} / \text{Total de funcionalidades del módulo} * 100$ G_C : Grado de cohesión	Mientras más cercano a 100 más cohesionado
	Grado de acoplamiento	Determinar cuán acoplados se encuentran los módulos a nivel de funcionalidades.	$G_{AC} = \text{Cantidad de módulos llamados} + \text{cantidad de módulos que lo llaman} / \text{Total de módulos} * 100$.	Mientras más cercano a 100 más acoplado

			G _{AC} : Grado de acoplamiento	
Adaptabilidad	Grado de adaptabilidad sin implementación	Determinar cuán adaptable son los productos sin implementación.	$G_{ADS} = \text{cantidad de requisitos parametrizables posibles a lograr} / \text{Total de requisitos} * 100$ G _{ADS} : Grado de adaptabilidad sin implementación	Mientras más cercano a 100 más adaptable

Recolección de datos de la muestra

Para la recolección de datos de la muestra se revisó toda la documentación generada en la etapa de Desarrollo de Requisitos de cada uno de los productos y a partir de aquí se establecieron los valores iniciales de cada uno de los indicadores. Posteriormente se realizó una re-ingeniería de los productos seleccionados, realizando todas las actividades y artefactos planteados en esta investigación y se volvió a realizar las mediciones para establecer los valores finales de los indicadores establecidos. En la Figura 18 se muestran los resultados de la recolección de los datos para cada uno de los indicadores.

Nombre_Muestra	Grupos	Parametrización_inicial	Parametrización_Final	Acoplamiento_inicial	Acoplamiento_Final	Cohesión_inicial	Cohesión_Final
Banco	1	20.00	66.70	50.00	18.00	15.00	15.00
COPA	1	21.40	92.90	90.00	18.00	14.00	14.00
Capital Humano	1	19.20	30.80	70.00	18.00	26.00	26.00
Contabilidad	1	42.10	63.20	50.00	18.00	19.00	19.00
Estructura y composición	1	20.00	25.00	90.00	18.00	10.00	10.00
AFT	2	16.70	16.70	60.00	60.00	72.00	72.00
Costos y procesos	2	22.27	22.27	90.00	90.00	11.00	11.00
Caja	2	23.50	23.50	70.00	70.00	17.00	17.00
Inventario	2	10.10	10.10	60.00	60.00	119.00	119.00
Facturación	2	25.00	25.00	40.00	40.00	25.00	25.00

Figura 18 Recolección de datos de los indicadores

Diseño del experimento

Para el análisis de las variables se utilizarán las validaciones establecidas por Sampieri y otros en [38]. La variable independiente se manipulará su presencia como vía de estímulo al comportamiento de las variables dependientes.

La medición de las variables dependientes se realizará a través de los Test de Mann Whitney y Wilcoxon. Para el caso del indicador adaptabilidad con implementación se aplicará un preexperimento, en su variante de preprueba y posprueba.

El experimento quedó conformado por dos grupos, uno de control (Grupo 2) y uno experimental (Grupo 1) cada uno con 5 productos seleccionados de forma aleatoria. En la Tabla 9 se muestra la asignación de estos grupos.

Tabla 9 Asignación aleatoria de grupos a los productos seleccionados en la muestra

Productos	Solución modular, integral y adaptable	Fase concluida	Grupo
Contabilidad	Sí	Certificación	1
Banco	Sí	Certificación	1
Cobros y Pagos	Sí	Certificación	1
Capital Humano	Sí	Certificación	1
Estructura y Composición	Sí	Certificación	1
Costos y Procesos	Sí	Certificación	2
Caja	Sí	Certificación	2
Inventario	Sí	Certificación	2
Facturación	Sí	Certificación	2
Activo Fijo Tangible	Sí	Pruebas internas	2

Tomando como principio que la variable independiente de esta investigación toma valores de presencia (Sí) cuando se aplica el proceso de Desarrollo de Requisitos modificado y ausencia (No) cuando se aplica el proceso de Desarrollo de Requisitos tradicional, entonces se somete a verificación el estado de la variable dependiente en ambos casos.

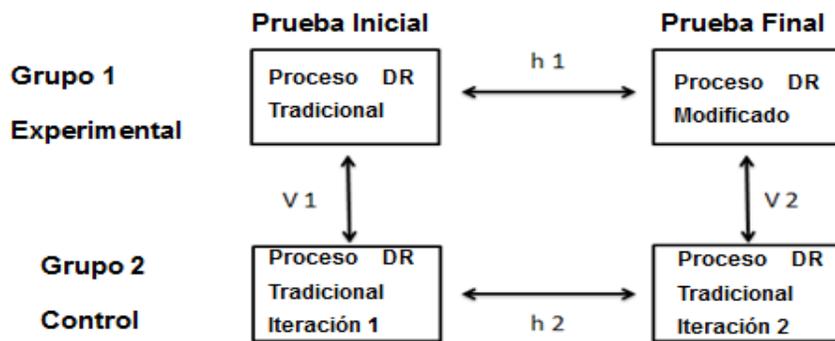


Figura 19 Diseño del experimento

Ejecución del Experimento

Ejecución del experimento para v1: Primeramente se realiza una evaluación de los dos grupos en (v1), con el objetivo de verificar la existencia o no de diferencias significativas entre ellos. Como se muestra en la Figura 20 los resultados fueron mayores de 0.05, por lo que las diferencias entre los grupos se consideran no significativas, lo cual se interpreta como que ambos grupos están en igualdad de condiciones una vez que comienza a ejecutarse el experimento.

Mann-Whitney Test

Ranks				
	Grupos	N	Mean Rank	Sum of Ranks
Param_Inicial	1	5	5.60	28.00
	2	5	5.40	27.00
	Total	10		
Acop_Inicial	1	5	5.90	29.50
	2	5	5.10	25.50
	Total	10		
Coh_Inicial	1	5	4.40	22.00
	2	5	6.60	33.00
	Total	10		

Test Statistics ^a				Param_Inicial	Acop_Inicial	Coh_Inicial
Mann-Whitney U				12.000	10.500	7.000
Wilcoxon W				27.000	25.500	22.000
Z				-.105	-.427	-1.149
Asymp. Sig. (2-tailed)				.917	.669	.251
Exact Sig. [2*(1-tailed Sig.)]				1.000 ^a	.690 ^a	.310 ^a
Monte Carlo Sig. (2-tailed)	Sig.			.968 ^b	.738 ^b	.309 ^b
	99% Confidence Interval	Lower Bound		.968	.737	.308
	Upper Bound			.969	.740	.310
Monte Carlo Sig. (1-tailed)	Sig.			.484 ^b	.369 ^b	.154 ^b
	99% Confidence Interval	Lower Bound		.483	.367	.153
	Upper Bound			.485	.370	.155

Figura 20 Análisis de resultados para V1

Ejecución del experimento para h2: Luego al grupo control en (h2) se le realiza una evaluación del Desarrollo de Requisitos en dos iteraciones del ciclo de vida de los productos, para verificar si su comportamiento se mantiene igual o no en distintos momentos del desarrollo. Como se muestra en la Figura 21 los resultados son mayores a 0.05, por lo que no se consideran significativos por lo que se puede interpretar que para todos los indicadores, los grupos mantienen su comportamiento en distintas etapas del desarrollo.

Wilcoxon Signed Ranks Test

		N	Mean Rank	Sum of Ranks
Param_Final - Param_Inicial	Negative Ranks	0	.00	.00
	Positive Ranks	0	.00	.00
	Ties	5		
	Total	5		
Acop_Final - Acop_Inicial	Negative Ranks	0	.00	.00
	Positive Ranks	0	.00	.00
	Ties	5		
	Total	5		
Coh_Final - Coh_Inicial	Negative Ranks	0	.00	.00
	Positive Ranks	0	.00	.00
	Ties	5		
	Total	5		

Test Statistics^a

	Param_Final - Param_Inicial	Acop_Final - Acop_Inicial	Coh_Final - Coh_Inicial
Z	.000 ^a	.000 ^a	.000 ^a
Asymp. Sig. (2-tailed)	1.000 ^b	1.000 ^b	1.000 ^b
Exact Sig. (2-tailed)	1.000 ^b	1.000 ^b	1.000 ^b
Exact Sig. (1-tailed)	1.000 ^b	1.000 ^b	1.000 ^b
Point Probability	1.000 ^b	1.000 ^b	1.000 ^b

Figura 21 Análisis de resultados para h2

Ejecución del experimento para h1: Una vez establecido esto al grupo experimental en (h1) se le aplica una evaluación antes de aplicadas las modificaciones al proceso de Desarrollo de Requisitos y una evaluación luego de aplicado este proceso modificado, con el objetivo de verificar si los cambios realizados en el proceso representan diferencias significativas. Como se muestra en la Figura 22 para los indicadores de parametrización y acoplamiento los valores fueron por debajo de 0.05 por lo que se considera que los cambios realizados al proceso de Desarrollo de Requisitos constituyen cambios significativos. En el caso del indicador de cohesión no se obtuvo

cambios significativos, lo cual se considera positivo bajo el principio de mantener una alta cohesión y un bajo acoplamiento entre componentes para obtener un buen diseño.

Wilcoxon Signed Ranks Test

		N	Mean Rank	Sum of Ranks
Param_Final - Param_Inicial	Negative Ranks	0	.00	.00
	Positive Ranks	5	3.00	15.00
	Ties	5		
	Total	10		
Acop_Final - Acop_Inicial	Negative Ranks	5	3.00	15.00
	Positive Ranks	0	.00	.00
	Ties	5		
	Total	10		
Coh_Final - Coh_Inicial	Negative Ranks	0	.00	.00
	Positive Ranks	0	.00	.00
	Ties	10		
	Total	10		

Test Statistics^{e,f}

			Param_Final - Param_Inicial	Acop_Final - Acop_Inicial	Coh_Final - Coh_Inicial
Z			-2.023 ^a	-2.041 ^b	.000 ^c
Asymp. Sig. (2-tailed)			.043	.041	1.000
Monte Carlo Sig. (2-tailed)	Sig.		.063	.062	
	99% Confidence Interval	Lower Bound	.062	.062	
		Upper Bound	.063	.063	
Monte Carlo Sig. (1-tailed)	Sig.		.032	.031	
	99% Confidence Interval	Lower Bound	.031	.031	
		Upper Bound	.032	.032	
Exact Sig. (2-tailed)					1.000 ^d
Exact Sig. (1-tailed)					1.000 ^d
Point Probability					1.000 ^d

Figura 22 Análisis de resultados para h1

Ejecución del experimento para v2: Por último en (v2) se verifica si en las evaluaciones realizadas en los estados finales de ambos grupos se observan o no diferencias significativas. Como se muestra en la Figura 23 los resultados para los indicadores de parametrización y acoplamiento son significativos, no siendo así para el indicador de cohesión, pues ningún módulo experimentó cambios en la cohesión de sus funcionalidades. Esto como se explicaba en el experimento anterior se considera positivo bajo el principio de diseño de mantener en los diseños alta cohesión y bajo acoplamiento. Con este último experimento se puede concluir que las modificaciones realizadas al proceso de Desarrollo de Requisitos constituirán cambios significativos en el grupo control donde no fue aplicado el experimento.

Mann-Whitney Test

	Grupos	N	Mean Rank	Sum of Ranks
Param_Final	1	5	7.90	39.50
	2	5	3.10	15.50
	Total	10		
Acop_Final	1	5	3.00	15.00
	2	5	8.00	40.00
	Total	10		
Coh_Final	1	5	4.40	22.00
	2	5	6.60	33.00
	Total	10		

	Param_Final	Acop_Final	Coh_Final	
Mann-Whitney U	.500	.000	7.000	
Wilcoxon W	15.500	15.000	22.000	
Z	2.511	2.705	1.110	
Asymp. Sig. (2-tailed)	.012	.005	.251	
Exact Sig. [2* (1-tailed Sig.)]	.008 ^a	.008 ^a	.310 ^a	
Monte Carlo Sig. (2-tailed)	Sig. .016 ^b	.008 ^b	.310 ^b	
	99% Confidence Interval	Lower Bound .016	.008	.308
		Upper Bound .016	.008	.311
Monte Carlo Sig. (1-tailed)	Sig. .008 ^b	.004 ^b	.155 ^b	
	99% Confidence Interval	Lower Bound .008	.004	.154
		Upper Bound .008	.004	.155

Figura 23 Análisis de resultados para v2

Una vez realizada cada evaluación, en todas se observan los resultados esperados por lo que se puede afirmar que las modificaciones realizadas al proceso de Desarrollo de Requisitos representarán diferencias significativas a la hora de construir nuevos productos que contengan características similares. En las Figuras 24 y 25 se muestran los resultados en forma de gráfica para los indicadores de parametrización y de acoplamiento. No se grafican los resultados de cohesión, pues su comportamiento siempre se mantuvo constante en el experimento.

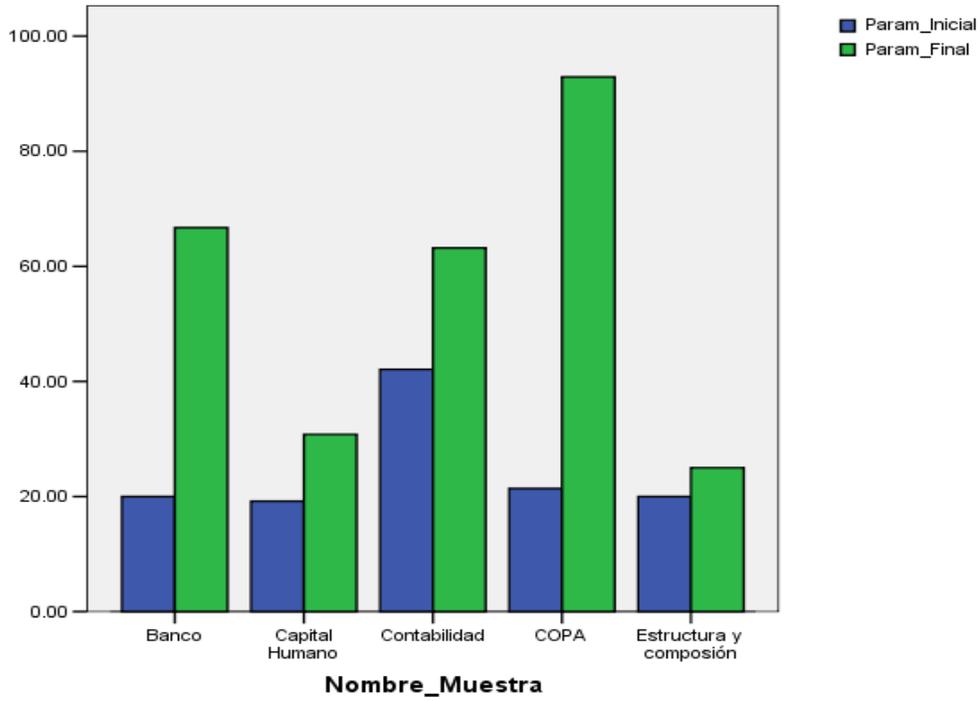


Figura 24 Gráfica de resultados para parametrización

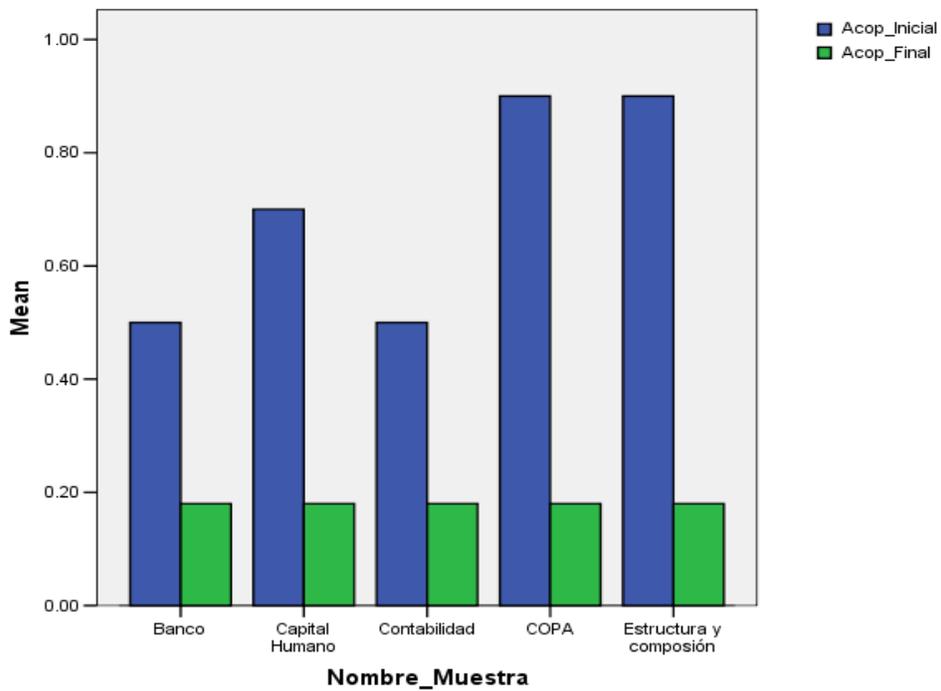


Figura 25 Gráfica de resultados para acoplamiento

Indicador adaptabilidad por implementación

El producto que se toma como base para evaluar la adaptabilidad por implementación es el Sistema de Mantenimiento Vehicular para la República Bolivariana de Venezuela. Este producto debía integrar parte de soluciones previamente construidas, tales como Capital humano, Activo Fijo y Control de Almacenes, más desarrollar las especificidades del Mantenimiento Vehicular. Por no existir una forma para Desarrollar Requisitos orientado a reutilización, esta solución se desarrolló desde cero, sin tener en cuenta que se podían reutilizar algunos elementos como los requisitos.

Para evaluar la adaptabilidad por implementación posible a lograr se realizó primeramente una pre prueba donde se revisaron las especificaciones de requisitos de este producto y se observó realizando analogías de conceptos la existencia de funcionalidades candidatas a reutilizar de las cuales no se reutilizó ninguna por lo que la adaptabilidad inicial de este producto fue cero. Luego se realizó una re-ingeniería del producto aplicando los elementos propuestos en esta investigación. Posteriormente se aplicó una pos prueba y se observó que efectivamente existían 25 funcionalidades que podían haber reutilizado otras existentes. En la Tabla 10 se muestra una asociación de las funcionalidades del Sistema de Mantenimiento que pudieron ser reutilizados de otros productos.

Tabla 10 Similitudes detectadas con la aplicación del proceso de Desarrollo de Requisitos modificado

Número	Funcionalidades de Mantenimiento	Funcionalidades candidatas a reutilizar	Referencia
1	Gestionar documentos técnicos	Gestionar documentos	Configuración General de CedruX
2	Gestionar grupo de unidades	Gestionar grupo de AFT	Activo Fijo Tangible
3	Gestionar propiedades	Gestionar propiedades de AFT	Activo Fijo Tangible
4	Actualizar recursos	Gestionar productos al	Inventario

	utilizados a la Orden	documento	
5	Asociar herramientas utilizadas en la Orden de trabajo	Gestionar productos al documento	Inventario
6	Asociar repuestos utilizados en la Orden de trabajo	Gestionar productos al documento	Inventario
7	Gestionar Informe de baja de las unidades policiales	Gestionar Informe de baja de productos	Inventario
8	Gestionar informes	Gestionar documentos	Inventario
9	Gestionar memorándum	Gestionar documentos	Inventario
10	Gestionar órdenes de salida	Gestionar documento de salida	Inventario
11	Gestionar órdenes de trabajo	Gestionar documentos	Inventario
12	Quitar herramientas utilizadas en la Orden de trabajo	Gestionar productos al documento	Inventario
13	Quitar recursos humanos utilizados en la Orden de trabajo	Gestionar productos al documento	Inventario
14	Quitar repuestos utilizados en la Orden de trabajo	Gestionar productos al documento	Inventario

15	Gestionar accesorios a unidad	Gestionar partes de activo	Activo Fijo Tangible
16	Gestionar inspección técnica a la unidad	Gestionar inventario	Inventario, Activo Fijo
17	Gestionar unidades	Gestión Activo Fijo	Activo Fijo
18	Imprimir acta de asignación de accesorios	Imprimir Modelo de movimiento de activos	Activo Fijo
19	Imprimir acta de asignación de unidades	Imprimir Modelo de movimiento de activos	Activo Fijo
20	Imprimir acta de recepción de accesorios	Imprimir Modelo de Alta de activos fijos	Activos Fijos
21	Imprimir acta de recepción de unidades	Imprimir Modelo de Alta de activos fijos	Activos Fijos
22	Gestionar accesorios	Gestionar Activo Fijo (útiles)	Activos Fijos
23	Gestionar causas de fallas	Gestionar Motivo	Activos Fijos
24	Gestionar unidades de medida	Gestionar unidades de medida	Configuración General
25	Gestionar recursos humanos	Gestionar Persona	Capital humano

Impacto social

A partir de la elevación de los grados de parametrización y de modularidad de los productos en la etapa de Desarrollo de Requisitos se contribuirá a la elevación de estos indicadores de manera global en el sistema. Esto permitirá formar mayor cantidad de productos y por tanto se elevará la posibilidad de insertarse en mayor de mercados tanto nacionales como extranjeros. En el caso del ERP-Cuba antes se podían formar solo los siguientes productos:

1. Estructura, contabilidad
2. Estructura, contabilidad, costos
3. Estructura, Contabilidad, Costos, Capital Humano
4. Estructura, Contabilidad, Costos, Capital Humano, Finanzas(Caja, banco, COPA)
5. Estructura, Contabilidad, Costos, Capital Humano, Finanzas(Caja, banco, COPA), Logística (Inventario, Facturación)
6. Estructura, Contabilidad, Costos, Capital Humano, Finanzas(Caja, banco, COPA), AFT

Luego de aplicar la propuesta solamente por el desacoplamiento funcional se pueden formar tantos productos como dominios existan implementados en este caso 10, así como tantas combinaciones puedan realizarse entre ellos.

Mediante el análisis de parametrizaciones se podrá aumentar la adaptabilidad en relación directa con la cantidad de parametrizaciones posibles a lograr y mediante la reutilización se podrán realizar productos con mayor facilidad ya que se utilizarán artefactos ya probados en un producto para la construcción de otros nuevos.

Conclusiones parciales

Luego de la realización de este capítulo se puede concluir lo siguiente:

- La modularidad lograda está basada en un diseño Aceptable, pues se mantuvo el principio de alta cohesión y bajo acoplamiento entre las funcionalidades que interrelacionaban los módulos.
- La adaptabilidad lograda también se considera Aceptable pues todos los resultados observados fueron positivos.

CONCLUSIONES GENERALES

- La genericidad del proceso tradicional de Ingeniería de Requisitos permitió la inclusión de nuevos artefactos, roles y actividades, así como el análisis sobre otras actividades existentes.
- Se podrán construir sistemas modulares, integrales y adaptables con un proceso de Desarrollo de Requisitos ajustado a sus características distintivas.
- Se pudo comprobar con la aplicación y evaluación del proceso de Desarrollo de Requisitos modificado, que se aumenta el grado de adaptabilidad y modularidad de los productos en la etapa de Desarrollo de Requisitos.

RECOMENDACIONES

- Se recomienda desarrollar una herramienta como soporte automatizado a la solución propuesta, en aras de agilizar el proceso de Desarrollo de Requisitos.
- Se recomienda añadir al proceso de Desarrollo de Requisitos modificaciones para lograr aumentar el grado de integralidad del sistema.

REFERENCIAS BIBLIOGRÁFICAS

1. **Atem De Carvalho, Rogério, De Campos, Renato.** *Uma análise de aspectos relacionados ao desenvolvimento e adoção de Enterprise Resources Planning livre de código aberto.* São Carlos, Gest. Prod, 2009. v. 16
2. **Da Conceição Menezes, Paulo André, González-Ladrón-De-Guevara, Fernando.** *Maximización de los beneficios de los sistemas ERP.* Journal of Information Systems and Technology Management, 2010, Vol. 7(No. 1)
3. **M. Esteves, Jose, Pastor, Joan A. .** *An ERP Life-cycle-based Research Agenda, Universidad Politécnica de Cataluña,* 1999.
4. **Carreón Suarez Del Real, María Cristina.** *Construcción de un catálogo de patrones de requisitos funcionales para ERP.*[Tesis de Maestría]. España, Universidad Politécnica de Cataluña, 2008.
5. **Lage Codorníu, César.** *Solución arquitectónica de la Configuración General de Cedrux para la parametrización de negocio del sistema.*[Tesis de Maestría]: Centro de Informatización de la Gestión de Entidades. La Habana, Universidad de las Ciencias Infomáticas, 2010.
6. **Del Toro Ríos , Dr.José Carlos.** *Documento de Visión del programa ERP-Cuba.* La Habana, 2008.
7. **Sommerville, Ian.** *Ingeniería de requisitos* 7^{ma} ed. Madrid, Pearson Educación, 2005.
8. **Wieggers, Karl E.** *More about Software Requiremented.* Washington, Microsoft Press, 2006. ISBN:0735622671
9. **Pressman, Dr. Roger S.** *Software Engineering A Practitioner's Approach* 7^{ma} ed. Nueva York, McGraw-Hill, 2010.
10. **Soto López, Nilet M.** *Gestión de requisitos aplicada a proyectos para el desarrollo de sistemas de gestión policial.*[Tesis de Maestría]: Facultad 2. La Habana, Universidad de las Ciencias Informáticas, 2010.
11. **Valdés Jiménez, Sasha.** *Desarrollo de Requisitos para proyectos del dominio de la Gestión policial.*[Tesis de Maestría]: Facultad 2. La Habana, Universidad de las Ciencias Informáticas, 2010.
12. **Hernández León, Dr.Rolando A , González Coello,Sayda.** *El paradigma cuantitativo de la investigación científica.* La Habana, Editorial Universitaria, 2002.
13. **Brooks, F. P.** *The Mythical Man–Month,* 1995.

-
14. **IEEE.** *IEEE Guide for developing System Requirements Specifications Std 1233a-1998.* Institute of Electrical and Electronics Engineers 1998.
 15. **Sommerville, Ian.** *Ingeniería de requisitos* 8^{va} ed. Madrid, Pearson Educación, 2007.
 16. **Pérez Teruel, Karina.** *Modelo de Referencia para la Ingeniería de requisitos en proyectos de Bioinformática (INeR).*[Tesis de Maestría]: Facultad 2. La Habana, Universidad de las Ciencias Informáticas, 2007.
 17. **Team, Cmmi Product.** *CMMI for Development, version 1.2ed.* Pittsburgh, Carnegie Mellon University, 2006.
 18. **Díaz, Oscar.** *Líneas de productos de software.* España, Universidad del país Vasco, 2007.
 19. **López, Oscar, Laguna, Miguel Ángel, Marqués, José Manuel.** *Reutilización del Software a partir de Requisitos Funcionales en el Modelo de Mecano: Comparación de Escenarios.* España, 2000.
 20. **SEI.** *Software Engineering Institute, University of Carnegie Mellon,* Citado el: 10 de octubre de 2011]. Disponible en: <http://www.sei.cmu.edu/>
 21. **Mellado, D, Fernández-Medina, E, Piattini, M.** *Aplicando un Proceso de Ingeniería de requisitos de Seguridad de Dominio para Líneas de Producto Software* IEEE LATIN AMERICA TRANSACTIONS, 2008, Vol.6(No.3).
 22. **Rodríguez, Juan, González, Sebastián.** *Líneas de productos de software.*[Tesis de Maestría]: Facultad de Ingeniería. Montevideo, Universidad de la República de Uruguay, 2007.
 23. **Montilva, Jonás A.** *Desarrollo de Software Basado en Líneas de Productos de Software.* 2006.
 24. **Española, Real Academia.** *Diccionario de la Real Academia Española,* [Sitio Web]. Citado el: 21 de abril de 2011]. Disponible en: <http://rae.es>
 25. **Torío Mellado, Julio.** *Estrategia de pruebas de líneas de productos de sistemas de tiempo real especificados con diagramas de estado jerárquicos.*[Tesis Doctoral]: Departamento de Ingeniería de Sistemas Telemáticos. España, Universidad politécnica de Madrid, 2004.
 26. **Nicolás Ros, Joaquín.** *Una propuesta de gestión integrada de requisitos y modelos de líneas de productos de software.*[Tesis Doctoral]: Facultad de Informática. España, Universidad de Murcia, 2009.

-
27. **Alexander, Christopher.** *A Pattern Language*, Oxford University Press.1997.
 28. **Cockburn, Alistair** *Writing Effective Use Case*. Addison-Wesley, 2000.
 29. **Durán Toro, A., Ruiz Cortés, A., Corchuelo Gil, R., Toro Bonilla, M. .** *Identificación de Patrones de Reutilización de Requisitos de Sistemas de Información*. España, Universidad de Sevilla, 2000.
 30. **Riba, Carles, Molina,Arturo.** *Ingeniería Concurrente: Una metodología integradora*. España, Universidad de Cataluña 2006.
 31. **Cuenca, LI. , Boza, A., Sanchís, M. .** *Estudio comparativo de paquetes ERP*. XII Congreso de Ingeniería de Organización, 2008.
 32. **De La O-Ramos, R., Borja-Ramírez V., López-Parra,M.,Ramírez-Reivich,A.C.** *Método de diseño para variedad de productos basado en el análisis de su modularidad*. 2008, Vol.XI(Núm.1):
 33. **De La Peña Herrador, Blanca , Aguayo González,Francisco , Lama Ruiz, Juan Ramón, Del Pozo Madroñal,Nicolás.** *Diseño de plataformas modulares para productos*. Técnica industrial, 2010.
 34. **Lorca Fernández, Pedro, De Andrés Suárez,Javier.** *Efectos de la implantación de sistemas integrados de gestión (ERP) en las grandes empresas españolas*. España, 2007. Vol. XXXVI
 35. **Núñez, R. Johana Del V, Rodríguez, C. De Los a,Arianny.** *Relación Costo-Beneficio que ofrece el Sistema Integrado de Gestión de Información SAP/R3 y el proceso de resistencia al cambio en el departamento de la empresa CADAPE Región I Sucre-Cumaná*. [Tesis de Maestría]: Departamento de contaduría. Cumaná, Universidad de Oriente Núcleo de Sucre, 2008.
 36. **Martín , César.** *Gestión y modelado de procesos con SAP NetWeaver BPM*. Mallorca, 2010.
 37. **Woods, Dan , Word ,Jeff.** *SAP NetWeaver For Dummies*. Canadá, Wiley Publishing, 2004.
 38. **Hernández Sampieri, Roberto, Fernández Collado,Carlos,Baptista Lucio, Pilar.** *Metodología de la Investigación*4^{ta} ed. México, McGrawGill, 2006. 970-10-5753-8

ANEXOS

Anexo 1 Especificaciones de los comportamientos de los requisitos típicos

Precondiciones

<<La precondición declara de qué debe estar seguro el sistema antes de autorizar el inicio del caso de uso. Esto no debe ser chequeado de nuevo mientras dura el caso de uso. Usualmente tener una precondición indica que otro caso de uso está o se ha ejecutado. Una precondición debe escribirse como una afirmación simple en pasado continuo. Por ejemplo: El cliente ha sido validado. (COCKBURN '00). >>

Flujo de eventos

Flujo básico

Se introducen los datos de<<|>> <<concepto>>

<<atributo 1>>

<<atributo 2>>

El sistema valida (ver validación <<número de la validación>>) los datos introducidos.

Si los datos son correctos el sistema los registra.

El sistema confirma el registro de los datos.

Concluye el requisito.

Pos-condiciones

Se registró en el sistema un<<a>> nuev<<o,a>> <<concepto>>.

Flujos alternativos

Flujo alternativo 4.a Información errónea

El sistema señala los datos erróneos y permite corregirlos.		
El usuario corrige los datos.		
Volver al paso <<número del paso en el que se hace la validación>> del flujo básico.		
Pos-condiciones		
N/A		
Flujo alternativo 4.b Información incompleta		
El sistema señala los datos vacíos y permite corregirlos.		
El usuario corrige los datos.		
Volver al paso <<número del paso en el que se hace la validación>> del flujo básico.		
Pos-condiciones		
N/A		
Flujo alternativo *.a El usuario cancela la acción		
Concluye el requisito.		
Pos-condiciones		
No se registran los datos.		
Validaciones		
Se validan los datos según lo establecido en el Modelo conceptual <<Referencia al modelo conceptual en cuestión>>.		
Relaciones	Requisitos Incluidos	Listado de los requisitos incluidos por el requisito. <<Evento en el que se incluye>>: <<Nombre del

		<p>requisito>>, en la agrupación <<Nombre de la agrupación de requisitos>>.</p> <p>Ejemplo: Seleccionar Activos Fijos: Adicionar activos fijos, en la agrupación Gestionar activos fijos.</p>
	Extensiones	<p>Requisitos que extienden a este requisito.</p> <p><<Evento en el que se extiende>>: <<Nombre del requisito>>, en la agrupación <<Nombre de la agrupación de requisitos>>.</p> <p>Ejemplo: Seleccionar módulo: Adicionar activos fijos, en la agrupación Gestionar activos fijos.</p>
Conceptos	<<Nombre del concepto>>	<p>Atributos del concepto que se utilizan en el requisito, tanto internamente como para mostrarlos al usuario. Deben separarse en dos grupos:</p> <p>Visibles en la interfaz:</p> <p><<Nombre del atributo 1>></p> <p>Utilizados internamente:</p> <p><<Nombre del atributo 2>></p>
Requisitos especiales		<p>Son los requisitos no funcionales específicos para el requisito.</p>
Asuntos pendientes		<p>Posibles mejoras al requisito.</p>

Anexo 2 Plantilla Base para la descripción de requisitos (parte fija del patrón)

Precondiciones	
Flujo de eventos	
Flujo básico	

Pos-condiciones

Flujos alternativos

Pos-condiciones

Validaciones

Relaciones Requisitos
 Incluidos

Extensiones

Conceptos

Requisitos
especiales

Asuntos
pendientes

Plantilla para describir las parametrizaciones o variabilidades (Parte variable)

Precondiciones

Descripción

Conceptos

Validaciones

Relaciones Requisitos
 Incluidos

Parámetros

Requisitos especiales

Asuntos pendientes
