



**Universidad Nacional Mayor de San Marcos**  
Universidad del Perú. Decana de América  
Facultad de Ingeniería de Sistemas e Informática  
Escuela Académico Profesional de Ingeniería de Sistemas

**Elección de un modelo de calidad para la gestión de  
requisitos en colegios privados**

**TESINA**

Para optar el Título Profesional de Ingeniero de Sistemas

**AUTOR**

Guisella Clara ANGULO ARMIJO

**ASESOR**

Lic. Jorge CHÁVEZ SOTO

Lima, Perú

2010



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

## Referencia bibliográfica

---

Angulo, G. (2010). *Elección de un modelo de calidad para la gestión de requisitos en colegios privados*. Tesina para optar el título profesional de Ingeniero de Sistemas. Escuela Académico Profesional de Ingeniería de Sistemas, Facultad de Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima, Perú.

---

# **ELECCIÓN DE UN MODELO DE CALIDAD PARA LA GESTIÓN DE REQUISITOS EN COLEGIOS PRIVADOS**

**BR: GUISELLA CLARA ANGULO ARMIJO**

Tesina presentada a consideración del cuerpo docente de la facultad de Ingeniería de Sistemas e Informática de la Universidad Nacional Mayor de San Marcos como parte de los requisitos para obtener el título profesional de Ingeniería de Sistemas

Aprobado por:

---

**Mg. Luzmila Pro Concepción**  
**Presidenta**

---

**Lic. Ulises Román Concha**  
**Miembro Docente**

---

**Lic. Jorge Chávez Soto**  
**Miembro Asesor**

Lima – Perú  
(Junio - 2010)

## FICHA CATALOGRAFICA

ANGULO ARMIJO GUISELLA CLARA

ELECCIÓN DE UN MODELO DE CALIDAD PARA LA GESTIÓN DE REQUISITOS  
EN COLEGIOS PRIVADOS

(Lima, 2010)

XII, 80 p., 29.7 cm (UNMSM)

Ingeniería de Sistemas, 2010

Tesina, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e  
Informática.

1. Ingeniería de Sistemas

I. UNMSM/ FISI II.

ELECCIÓN DE UN MODELO DE CALIDAD PARA LA GESTIÓN DE REQUISITOS  
EN COLEGIOS PRIVADOS

## **DEDICATORIA**

A mis padres y a mi familia por sus apoyos constantes.

## **AGRADECIMIENTOS**

A mi familia mi motivo de salir adelante, sobre todo a mi Madre quien me enseña a diario que en esta vida todo se logra, solo con esfuerzo.

A mi asesor Lic. Jorge Chávez, por sus consejos y sus sugerencias para hacer realidad la finalización de la presente tesina.

A la UNMSM por haber sido mi casa de estudio en todos estos años de Pre-grado y brindarme una cálida acogida y la posibilidad de ser una hoy una profesional.

(“El ser excelente hace todo aquello que el mediocre no sería capaz de realizar y está convencido que solamente a través de su entrega incondicional y generosa el mundo puede mejorar, es protagonista del cambio, es el arquitecto social de su tiempo, el ser excelente es por supuesto un triunfador”)

*Miguel Ángel Cornejo*



## RESUMEN

### ELECCIÓN DE UN MODELO DE CALIDAD PARA LA GESTIÓN DE REQUISITOS EN COLEGIOS PRIVADOS

**Br:** Guisella Clara Angulo Armijo

**Asesor:** Lic. Jorge Chávez Soto

**Título obtenido:** Ingeniero de sistemas

---

*La presente tesina tiene como principal objetivo la elección de un modelo de calidad para la adecuada Gestión de Requisitos en los colegios privados de Lima. El Reporte de Caos ofrecido por “STANDISH GROUP” [1] muestra datos acerca de lo que ocurre con los proyectos de Software, señalando que los principales factores por los cuales los proyectos software fracasan son: La mala gestión de los requisitos y una mala comprensión inicial de los mismos.*

*Los Colegios no escapan de esta problemática en la Gestión de requisitos y para el cumplimiento del objetivo señalado se realizará una comparativa entre modelos de calidad (NTP ISO 12207, CMMI nivel 2 y METRICA 3) en base a factores seleccionados por el Método Scoring y criterios seleccionados en base a las similitudes entre los modelos de calidad, que serán apoyo fundamental para realizar la comparativa de los modelos de calidad y de esta forma determinar la mas optima.*

*La elección de un modelo de un Modelo de calidad reconocido internacionalmente, permitirá el seguimiento de las mejores prácticas comprobadas y asegurará una gestión de Requisitos adecuada en los colegios particulares de Lima, que su vez proporcionará la óptima entrada para las siguientes actividades en el proceso de desarrollo de Software.*

*Palabras Claves: Ingeniería de requisitos, Modelos de Calidad, ISO 12207, CMMI, METRICA 3*

## ABSTRACT

### ELECTION OF A MODEL OF QUALITY MANAGEMENT REQUIREMENTS IN PRIVATE SCHOOLS

**Br:** Guisella Angulo Clara Armijo

**Advisor:** Lic. Jorge Chavez Soto

**Qualification obtained:** Systems engineer

---

*This dissertation's main objective is the choice of a quality model for proper management requirements in private schools in Lima. The Chaos Report offered by "Standish Group [1] shows data about what happens to the software projects, noting that the main factors why software projects fail are: Poor management of requirements and a poor understanding the same initial.*

*The Colleges do not escape from this problem in the management requirements and to fulfil the stated objective will be a comparison of quality models (NTP ISO 12207, CMMI Level 2 and 3 METRIC) based on factors selected by the Scoring Method and Criteria selected based on the similarities between the models of quality and support will be essential for making the comparison of quality models and thus determine the most optimal.*

*The choice of a model of an internationally recognized quality model will allow monitoring of proven best practices and ensure proper management of requirements in the private schools of Lima, which in turn provide the best entry for the following activities in the process of Software development.*

Keywords: Requirements Engineering, Models, Quality, ISO 12207, CMMI, METRIC 3

## INDICE

	Pág.
DEDICATORIA.....	iv
AGRADECIMIENTOS .....	v
RESUMEN.....	vii
ABSTRACT .....	viii
Indice .....	9
INDICE DE TABLAS .....	xi
INDICE DE FIGURAS.....	xii
INTRODUCCION .....	13
1. CAPITULO I: Fundamentos de las TIC y Planteamiento del Problema.....	14
1.1 Antecedentes .....	14
1.1.1 Gestión de Proyectos Software [2].....	14
1.1.2 El rol de las TIC para alcanzar los objetivos internacionales para la educación [3].....	16
1.1.3 Una revisión crítica de Las TIC para el sector educativo .....	17
1.1.4 Desarrollo de Software en el Perú [4] .....	17
1.1.5 Factores de éxito de un proyecto Software [1].....	19
1.1.6 Proyectos software que fracasan .....	20
1.1.7 Consecuencias de los proyectos que fracasan .....	21
1.2 Planteamiento del Problema.....	23
1.2.1 Definición del problema.....	23
1.2.2 Objetivos generales y específicos .....	24
1.2.2.1 Objetivo general.....	24
1.2.2.2 Objetivos específicos .....	24
1.3 Alcance y Limitaciones del estudio .....	25
1.4 Justificación de la investigación.....	25
2. CAPITULO II: Gestión de Requisitos y Modelos de Calidad .....	26

2.1.	¿Qué es un Requisito? .....	26
2.2.	Características de los Requisitos .....	26
2.3.	Requisitos Funcionales y No funcionales: .....	27
2.4.	Ingeniería de Requisitos .....	30
2.5.	Proceso de la ingeniería de requisitos [8].....	30
2.6.	Actualidad en la Elicitación de Requisitos.....	34
2.7.	Herramientas para la Gestión de Requisitos en la actualidad .....	37
2.8.	Modelos de Calidad.....	38
2.9.	Métricas.....	48
2.10.	Estudios actuales en la mejorar la calidad en la Gestión de Requisitos... ..	52
3.	CAPITULO III: Definición del método de elección del Modelo de Calidad para mejorar la completitud de Requisitos en los colegios privados de Lima..	54
3.1.	Planteamiento de la Solución .....	55
3.2.	Definición del método de elección del Modelo de Calidad .....	55
3.3.	Análisis e interpretación de los resultados .....	59
3.4.	Factores de Evaluación.....	60
4.	CAPITULO IV: Desarrollo para la elección del Modelo de Calidad para la Gestión de Requisitos en colegios privados de Lima .....	61
4.1	Requisitos para desarrollo del caso de estudio.....	61
4.2	Problemática en la Institución educativa Bertolt Brecht.....	61
4.3	Aplicación del método de selección del Modelo de Calidad .....	63
4.4	Análisis e interpretación de los resultados .....	72
4.5	Factores de Evaluación aplicados al Método de Selección.....	73
5.	CONCLUSIONES.....	74
6.	RECOMENDACIONES.....	76
	BIBLIOGRAFIA.....	77
	ANEXO I. Diagrama de procesos.....	79
	ANEXO II. METODO del SCORING.....	80

## INDICE DE TABLAS

	Pág.
Tabla 1-1 Actividades desarrolladas por empresas desarrolladoras de Software [4].....	19
Tabla 2-1 Vinculación de las etapas del proceso con.....	41
Tabla 2-2 Tareas del establecimiento de requisitos .....	49
Tabla 3-1 Factores de Éxito relacionados con la Planificación .....	60
Tabla 4-1 Necesidades en la Gestión de Requisitos.....	64
Tabla 4-2 Establecimiento de Factores .....	66
Tabla 4-3 Tabla de Factores .....	67
Tabla 4-4 Selección de Criterios .....	68
Tabla 4-5 Criterios .....	69
Tabla 4-6 Tabla Comparativa I .....	70
Tabla 4-7: Tabla comparativa .....	71
Tabla 4-8 Aplicación de las preguntas .....	73

## INDICE DE FIGURAS

	Pág.
Figura 1-1 Distribución por tamaño de Empresa [1].....	18
Figura 1-2 Factores de Éxito de un Proyecto Software [1] .....	20
Figura 1-3 División de los Proyectos Software [1] .....	21
Figura 1-4 Sobre-costos en los proyectos Software [1] .....	22
Figura 1-5 Sobre-tiempos en los proyectos Software [1].....	22
Figura 1-6 Funcionalidad Deficiente en el Software [1].....	23
Figura 2-1 Tipos de Requisitos No funcionales [7] .....	28
Figura 2-2 Proceso de ingeniería de Requerimientos [8].....	30
Figura 3-1 Identificación de Necesidades en la Gestión de Requisitos .....	56
Figura 3-2 Factores que afectan la gestión de Requisitos .....	57
Figura 3-3 Factores que afectan la gestión de Requisitos .....	59
Figura 4-1 Calculo del Resultado.....	67
Figura 4-2 Puntuación obtenida por cada Modelo .....	72

## INTRODUCCION

En la actualidad los proyectos surgen por necesidades insatisfechas o por oportunidades que se pueden aprovechar, pero el mal planeamiento de un proyecto causa pérdidas grandes de recursos (tiempo y dinero) que generan un malestar mayor en vez de solucionarlo. Por este motivo se debe considerar que para una adecuada formulación y gestión de proyectos se necesita tener todos los elementos necesarios para tal formulación: herramientas disponibles, análisis del entorno, etc.

Los proyectos software presentan la misma problemática y con mayor índice de fracaso que los proyectos en otros rubros, un estudio realizado por Standish Group [1], revela cuales son las principales causas por los cuales los proyectos software fracasan: La mala gestión de los requisitos y una mala comprensión inicial de los mismos.

El sector educativo que desarrollo software para su uso como herramienta de apoyo en la educación, no es ajeno al problema de sistemas software que fracasan, debido a lo mencionado la selección de un Modelo de Calidad que ayude en la correcta gestión de los Requisitos permitirá utilizar practicas comprobadas y reducirá los plazos y costos de las fases de desarrollo siguientes e incrementa el índice de satisfacción.

La investigación incluye el desarrollo de los siguientes capítulos:

Capitulo I, donde se presenta los fundamentos de las TICs y el planteamiento del problema; Capitulo II, donde se presentan los conceptos teóricos que sustentan la investigación; Capitulo III, donde se presenta la Definición del Método de elección del Modelo de Calidad para mejorar la completitud de Requisitos en los colegios privados de Lima; Capitulo IV, donde se procede con el desarrollo del Método definido para la elección del Modelo de Calidad, seguidamente se continuará con las conclusiones y las recomendaciones acerca de los futuros trabajos.

## 1. CAPITULO I: FUNDAMENTOS DE LAS TIC Y PLANTEAMIENTO DEL PROBLEMA

### 1.1 Antecedentes

#### 1.1.1 Gestión de Proyectos Software [2]

- **Tipos de Proyectos:**

En proyectos de software se diferencia tres tipos diferentes de proyectos, desde el punto de vista de su gestión:

- **Proyectos nuevos:** Se analizan costos, tiempos y cantidad de personas. Es el caso más difícil de todos.
- **Replanteo de proyectos viejos:** Se afinan las metodologías de estimación. Es la principal fuente de información.
- **Extensiones o ampliaciones de un proyecto existente:** Es un caso intermedio donde se desea tener buena precisión en plazos y costos. Siempre se debe tener en cuenta que los tres casos presentan situaciones diferentes y no pueden ser tratados por igual.

- **El tamaño de los proyectos:**

Los proyectos de software son diferentes por la sola razón de su **tamaño**.

Por el momento no podemos dar criterios de diferenciación cuantitativa, pero debemos tener presente que existen tres categorías diferenciadas de proyectos con problemas diferentes cada una:

- Proyectos **pequeños:** consisten solamente en implementación. No tienen costos indirectos importantes.
- Proyectos **medianos:** es un caso intermedio entre los otros dos.
- Proyectos **grandes:** poseen implementación, pero hay más cosas.



Poseen gerencia de proyecto, control de calidad, capacitación de personal, hay un plan de mantenimiento, hay documentación importante para uso interno y externo. Se genera información para mercadeo.

Un error clásico en la Gestión de proyectos es no advertir la existencia de tres categorías diferentes. Todavía hoy se sigue pensando que la información o la experiencia adquirida en proyectos pequeños pueden servir para proyectos medianos o grandes. Este es uno de los orígenes de los resultados catastróficos en la gestión de proyectos.

La constitución del equipo de trabajo es uno de los mejores indicadores del caso que se considera. Así por ejemplo, los proyectos pequeños poseen:

- Menos de un año de tiempo de desarrollo.
- Menos de 25 meses – persona de esfuerzo total.
- Menos de 3 personas en el equipo de trabajo.

Por el contrario, los proyectos grandes poseen:

- Más de 3 años de tiempo de desarrollo.
- Más de 100 meses persona.
- Más de 10 personas en el equipo de trabajo.

Un equipo de trabajo de menos de 3 personas puede trabajar con una documentación informal, un equipo de más de 10 personas, durante varios años, no puede confiar en los contactos personales ni en la memoria de la gente. Es más, puede ocurrir que muchos de los que comienzan el proyecto sean reemplazados por otros, en plazos tan largos. Esta diferencia en el manejo interno de la información hace una gran distinción entre los dos extremos de tamaños de proyectos.

Es muy importante observar que la barrera de 3 años es una barrera dramática para la informática. En tres años hay cambios importantes en las plataformas. De acuerdo con cifras usualmente aceptadas (ley de Gordon Moore), la capacidad de almacenamiento y la velocidad de las plataformas **se multiplica por diez** en tres

años. Tenemos un orden de magnitud de crecimiento. Esto implica, para un proyecto grande, que:

- La plataforma para la cual diseñamos será obsoleta en el momento de entrega.
- La disponibilidad de memoria, la velocidad de procesamiento y el tamaño del almacenamiento, que al comenzar el proyecto puede ser una limitación del diseño, puede dejar de serlo al final del proyecto.

Este elemento no debe dejarse de tener en cuenta.

- Es posible que aparezcan nuevas tecnologías y nuevas herramientas de trabajo con las cuales no se contaba al comenzar el proyecto.
- Es posible que aparezcan nuevas exigencias con las cuales no se contaba al comenzar el proyecto.

Estas consideraciones muestran que hay diferencias profundas en el armado y realización de un proyecto grande. Este es uno de los temas centrales de la gestión de proyectos.

### **1.1.2 El rol de las TIC para alcanzar los objetivos internacionales para la educación [3]**

La introducción de las TIC como una herramienta en apoyo del sector educativo ha sido motivo de sustanciales discusiones desde fines de la década del 90. Hace diez años, la atención se centraba en la Educación Técnica y Profesional y en la capacitación del personal docente. Durante los últimos años, un número cada vez mayor de agencias internacionales de desarrollo han comenzado a adoptar todo el potencial de las TIC en apoyo del sector educativo.

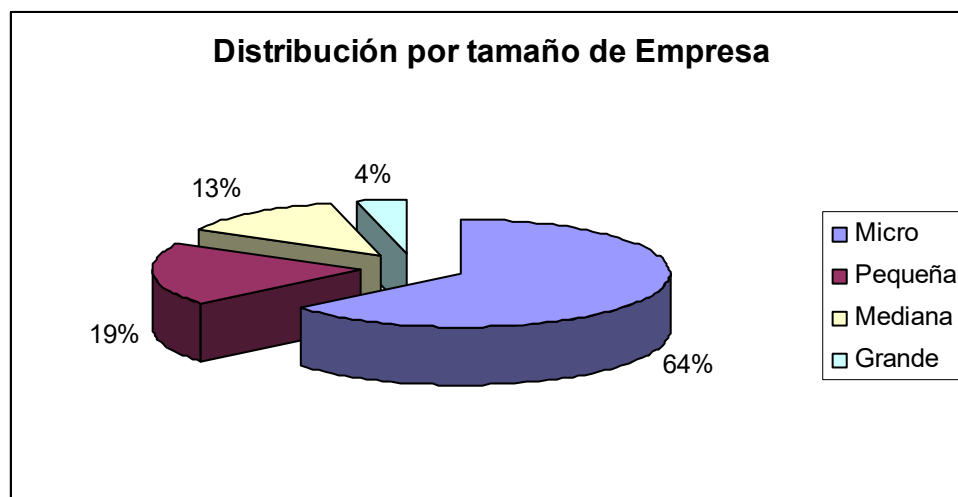
Cuando se analiza la integración de las TIC en apoyo del logro de objetivos educativos, es posible darse cuenta de que, luego de casi una década de usar las TIC para estimular el desarrollo, aún no se ha logrado su completa integración en las actividades para el desarrollo y que todavía será necesario continuar trabajando para incrementar la toma de conciencia al respecto. Aún así, durante la última década se han dado importantes avances.

### 1.1.3 Una revisión crítica de Las TIC para el sector educativo

- **Fines de la década del 70 – comienzos de los 80s: Programación:** Este fue el período en el que la razón pedagógica para el uso de la computadora era fundamentalmente la programación, ayudando al desarrollo de las habilidades lógicas y matemáticas de los estudiantes.
- **Fines de los 80s – principios de los 90s:** Aprendizaje basado en la Computadora: Cuando las computadoras multimedia, con gráficos y aplicaciones de sonido, comenzaron a estar disponibles, fueron usadas como apoyo en los procesos de enseñanza, en asignaturas básicas tales como las matemáticas, la lectura y la escritura.
- **Principios de los 90s: Aprendizaje basado en la Web:** La tercera ola en el uso de las TIC en el sector de la educación comenzó con el advenimiento de la World Wide Web. El uso de la Web es parcialmente producto de los inconvenientes que conlleva la actualización de contenidos por medio de CD-ROM. La información basada en la Web, por el contrario, permite que el contenido educativo sea actualizado con mucha frecuencia.
- **Fines de los 90s: Educación online:** La educación online o E-learning combina aplicaciones de aprendizaje basadas en la computadora con otras con base en la Web, en beneficio de los estudiantes. Ofrece también facilidades de interacción entre el docente y el estudiante y de los estudiantes entre si.

### 1.1.4 Desarrollo de Software en el Perú [4]

Se ha podido identificar a 150 empresas peruanas fabricantes de software a nivel de Lima Metropolitana, de las cuales más del 80% corresponde a la categoría de pequeñas y micro empresas. La figura 1-1 Distribución por tamaño de Empresa, muestra con los detalles la distribución.



**Figura 1-1 Distribución por tamaño de Empresa [1]**

El mercado del software es un sector relativamente joven y de alta especialización, cuyos niveles de inversión son relativamente menores comparados con otras industrias, pues su principal activo son los recursos humanos con los que cuenta.

En términos generales, una empresa típica de software tiene los siguientes procesos para desarrollar sus aplicaciones:

- Análisis de requerimientos
- Modelamiento de procesos
- Modelamiento de datos
- Desarrollo
- Prueba y entrega.

La tabla 1-1 Actividades desarrolladas por empresas desarrolladoras de Software, muestra que las empresas desarrolladoras de software no están muy especializadas en una sola actividad sino que se complementan con otras actividades de servicios para poder operar, destacando el desarrollo de soluciones a medida, la consultoría de sistemas, la fabricación de software, entre otras.

Actividades desarrolladas por empresas desarrolladoras de Software

**Tabla 1-1 Actividades desarrolladas por empresas desarrolladoras de Software [4]**

<b>Actividades</b>	<b>Porcentaje</b>
Desarrollador a medida	81,8 %
Consultaría de sistemas	79,1 %
Fabricante de software	78,4 %
Integrador de sistemas	66,2 %
Comercializador y distribuidor de software	62,2 %
Servicios informáticos diversos	61,5 %
Outsourcing	52,7 %
Servicios de Internet	35,1 %

### **1.1.5 Factores de éxito de un proyecto Software [1]**

El aspecto más importante de la investigación es descubrir por qué los proyectos fallan. Para ello, el Standish Group, los gerentes de TI encuestados sus opiniones acerca de por qué los proyectos tengan éxito. Las tres principales razones por las que un proyecto tenga éxito son la participación del usuario, el apoyo Gerencial, y una enunciación clara de los requisitos. Existen otros criterios de éxito, pero con estos tres elementos en su lugar, las posibilidades de éxito son mucho mayores. Sin ellos, probabilidad de fracaso aumenta dramáticamente, podemos observarlo en la Figura 1-2 Factores de Éxito de un Proyecto Software

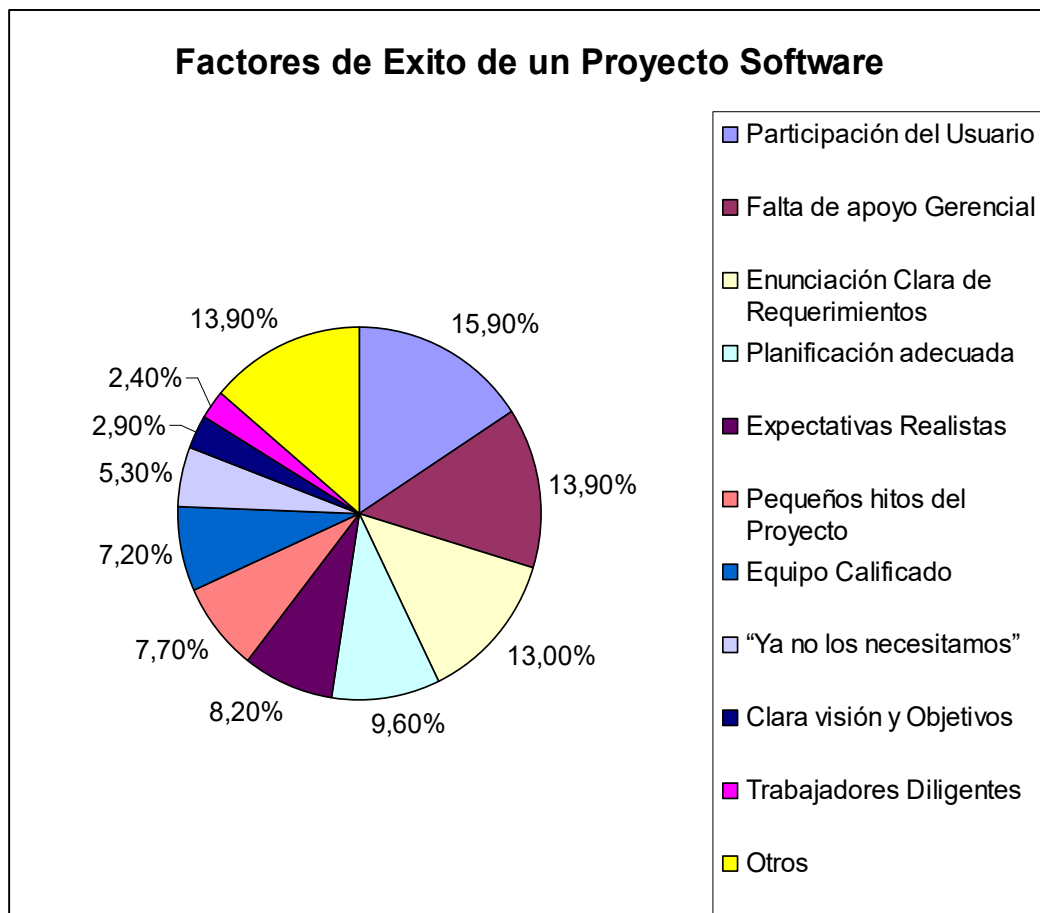


Figura 1-2 Factores de Éxito de un Proyecto Software [1]

### 1.1.6 Proyectos software que fracasan

El Reporte del Caos ofrecido por “STANDISH GROUP” divide los proyectos software en 3 categorías: Proyectos exitosos, proyectos cuestionados y proyectos dañados, señalando que el porcentaje de proyectos exitosos en promedio es de solo 16.2 %, mientras que los proyectos cuestionados son de 52,7% y el porcentaje de los proyectos dañados (cancelados) es de 31,1%.

Si consideramos el porcentaje de los proyectos exitosos (16,2 %) versus los no exitosos (68,9%) observamos que existen problemas graves que afectan el desenvolvimiento de los proyectos software, así como lo muestra la Figura 1-3 División de los Proyectos Software.

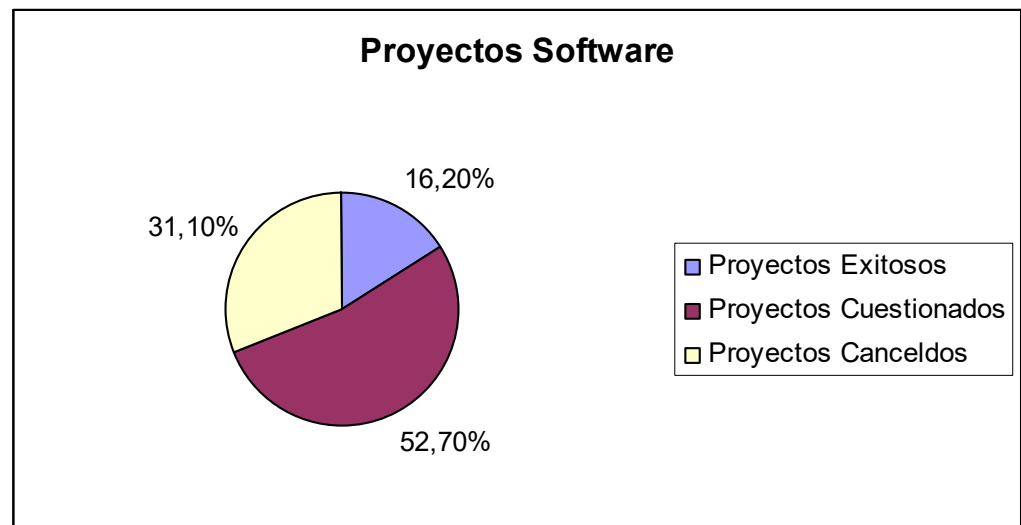
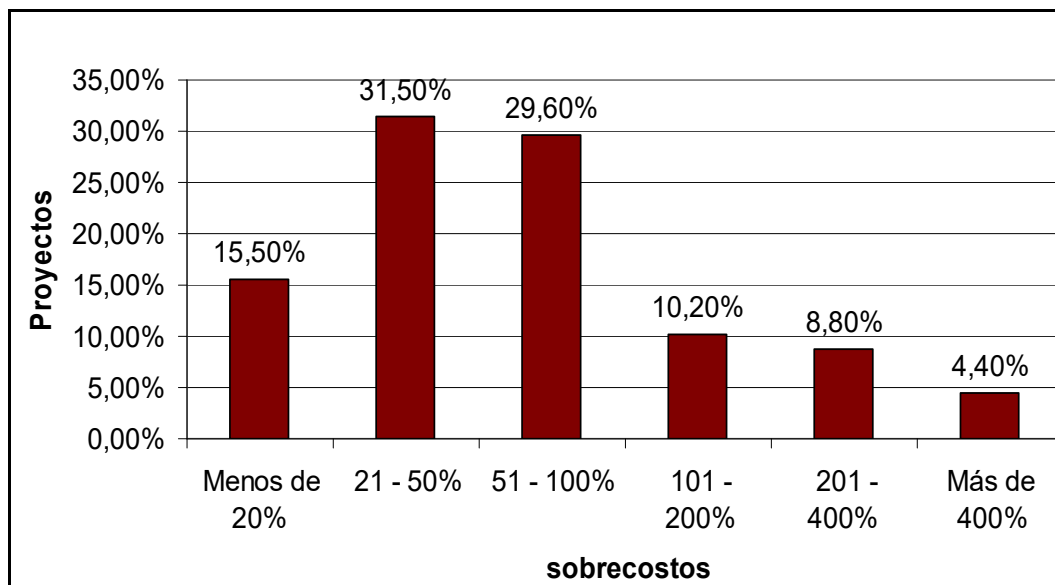


Figura 1-3 División de los Proyectos Software [1]

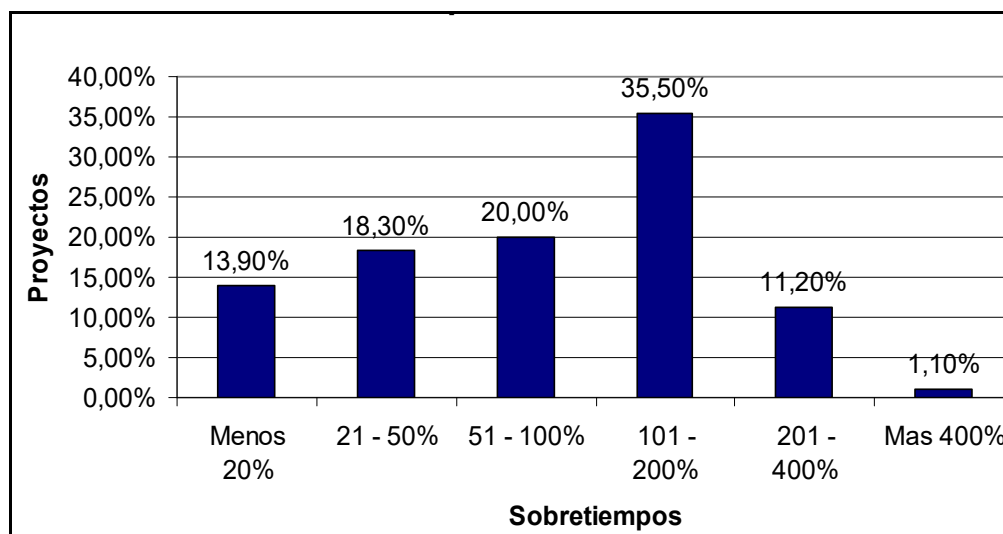
### 1.1.7 Consecuencias de los proyectos que fracasan

Los proyectos dañados causan: Sobre-costos, Sobre-tiempos y desarrollo de productos software con funcionalidad deficiente. En la Figura 1-4 Sobre-costos en los proyectos software, observamos los porcentajes de sobre-costos al que se incurre para completar los proyectos software. Casi un tercio experimenta un sobre costo de 150% a 200%.



**Figura 1-4 Sobre-costos en los proyectos Software [1]**

En la siguiente Figura 1-5 Sobre-tiempos en los proyectos Software, observamos los porcentajes de sobre Tiempo al que se incurre para completar los proyectos software. Más de un tercio de los proyectos experimentó sobre-costos de 200 a 300%.



**Figura 1-5 Sobre-tiempos en los proyectos Software [1]**

En la Figura 1-6 Funcionalidad Deficiente en el Software, observamos los porcentajes de Software que no cumplen con las funcionalidades para las cuales fueron creadas.



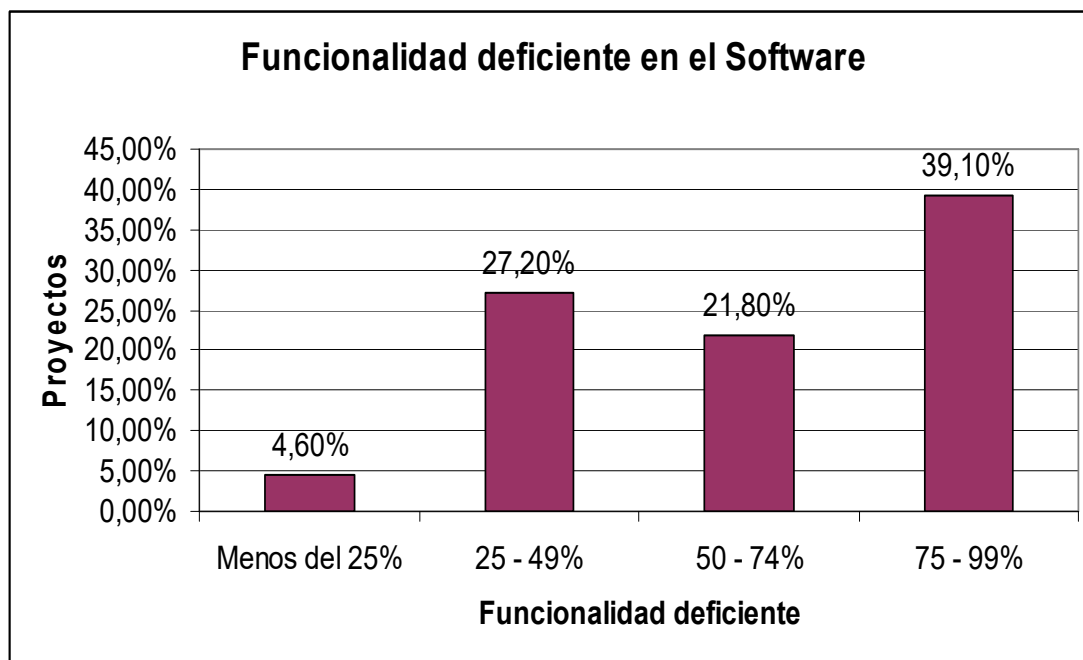


Figura 1-6 Funcionalidad Deficiente en el Software [1]

## 1.2 Planteamiento del Problema

### 1.2.1 Definición del problema

Sistemas Software ineficientes en colegios que pertenecen al sector educativo privado de Lima, que no cumplen con las restricciones de costos, tiempos y funcionalidad por los cuales fueron concebidos, debido a una mala gestión de requisitos.

Por ejemplo, en uno de los colegios particulares de Lima, la toma de requisitos se realiza a través de un documento de especificación de necesidades que es elaborado por el área solicitante del colegio y que es enviado al área de informática para su desarrollo, estos últimos revisan la especificación elaborando prototipos y estimando el tiempo que tomará el desarrollo así como los costos estimados, esta propuesta es enviada al área solicitante quien aprueba o desaprueba la propuesta, seguidamente se continua con el desarrollo del sistema.

La falta de técnicas para la gestión de requisitos produce a que no se planifique adecuadamente las actividades, continuando con el ejemplo anterior, al no planificar adecuadamente las validaciones, estas se realizan al terminar el desarrollo del sistema, produciendo que en la mayoría de los casos el sistema software no cumpla con las expectativas del área solicitante.

## **1.2.2 Objetivos generales y específicos**

### **1.2.2.1 Objetivo general**

Elegir un modelo de calidad para la adecuada Gestión de Requisitos en los colegios privados de Lima, utilizando para esto criterios definidos según las necesidades descritas, que permitirán la definición del modelo más óptimo para la realidad mencionada.

### **1.2.2.2 Objetivos específicos**

- Identificar las necesidades en la Gestión de Requisitos presentes en los colegios particulares privados
- Identificar los factores que afectan la Gestión de requisitos en los colegios privados, que ayuden a comprender las deficiencias que presentan en esa etapa tan crucial en el desarrollo de sistemas de información.
- Definir los criterios que sirva como guía en la definición de un modelo de calidad óptimo para el sector educativo privado de Lima.
- Seleccionar un Modelo de calidad óptimo para los colegios privados de Lima, utilizando para esto los factores y los criterios definidos con anterioridad.

### **1.3 Alcance y Limitaciones del estudio**

El alcance del estudio se restringe a colegios particulares de Lima, abarcando solo la forma en la que se gestiona la primera etapa en el proceso de desarrollo del software, es decir la etapa de captura, análisis y gestión de requisitos.

El estudio se limita a la evaluación de 3 modelos de calidad para la gestión de requisitos, permitiendo la elección del modelo más adecuado para la realidad de los colegios particulares.

El estudio se limitara a la revisión de los Modelos de calidad Métrica 3, CMMI nivel 2 e NTP ISO 12207, enfocando el tema de Gestión de Requisitos.

### **1.4 Justificación de la investigación**

El tema investigado es importante debido a que proporcionará la elección de un modelo de calidad para la gestión de requisitos, para su uso en colegios privados.

En la actualidad las instituciones educativas, no cuentan con un estándar o con procedimientos adecuados para la captura de requisitos y su gestión, provocando problemas en la identificación de los requisitos funcionales y no funcionales de los sistemas que se pretenden desarrollar, así como del planeamiento de los recursos que se utilizarán.

El tema seleccionado surge a raíz de la observación de las deficiencias en el área de desarrollo de sistemas de colegios particulares, específicamente problemas en la captura de requisitos.

Además considerando la importancia del papel de la educación en el desarrollo del Perú, una adecuada gestión de requisitos para la elaboración de sistemas software dentro de las instituciones educativas privadas contribuirá con el perfeccionamiento del sistema educacional, que responda a las necesidades demandadas por la sociedad.

## 2. CAPITULO II: GESTIÓN DE REQUISITOS Y MODELOS DE CALIDAD

### 2.1. ¿Qué es un Requisito?

La IEEE 610.12-1990[5] define requisito como:

- a) Una condición o capacidad requerida por un usuario para resolver un problema o alcanzar un objetivo.
  
- b) Una condición o capacidad que debe ser poseída por un sistema o componente del sistema para satisfacer un contrato, estándar, especificación, u otro documento formalmente impuesto.
  
- c) Una representación documentada de una condición o capacidad como las descritas en los incisos a) o b)

### 2.2. Características de los Requisitos

Un conjunto de requisitos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo [6].

- **Necesario:** Un requisito es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
  
- **Conciso:** Un requisito es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.

- **Completo:** Un requisito está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- **Consistente:** Un requisito es consistente si no es contradictorio con otro requisito.
- **No ambiguo:** Un requisito no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
- **Verificable:** Un requisito es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas.

### 2.3. Requisitos Funcionales y No funcionales:

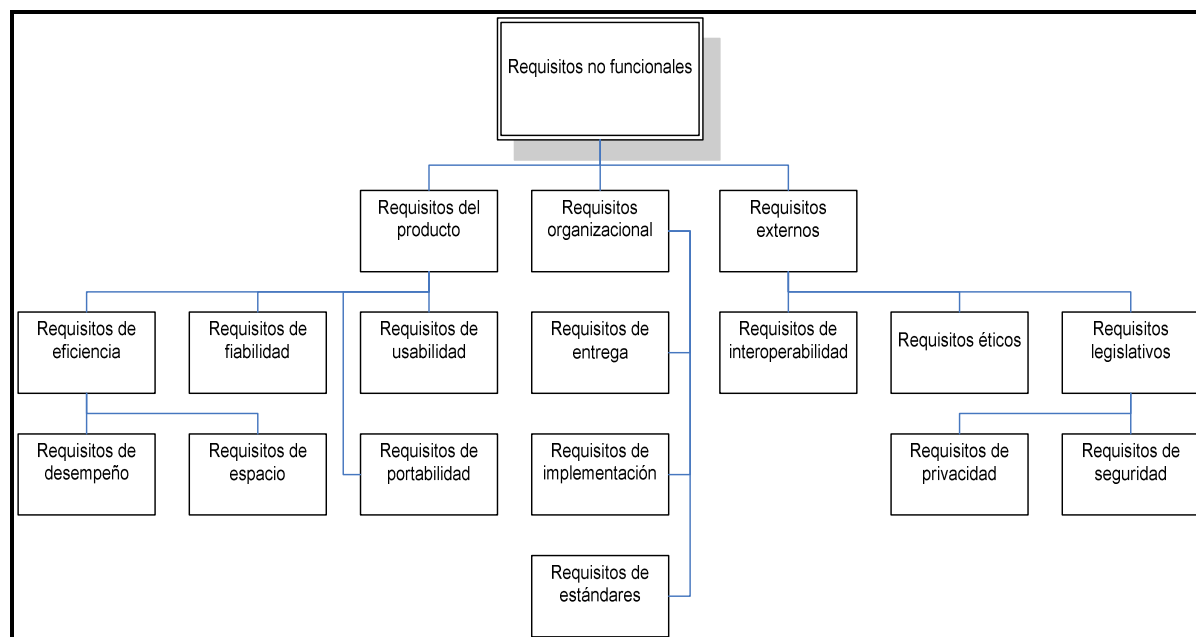
Somerville, Ian [7] presenta la siguiente clasificación de Requisitos divididos en:

- **Requisitos Funcionales**

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requisitos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer.

- **Requisitos No funcionales**

Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requisitos no funcionales a menudo se aplican al sistema en su totalidad. Normalmente apenas se aplican a características o servicios individuales del sistema. En la figura 2-1 Tipos de Requisitos No funcionales, muestra los tipos de requisitos no funcionales.



**Figura 2-1 Tipos de Requisitos No funcionales [7]**

- **Requisitos de Dominio**

Son requisitos que provienen del dominio de aplicación del sistema y que reflejan las características y restricciones de ese dominio. Pueden ser funcionales o no funcionales.

En realidad, la distinción entre diferentes tipos de requisitos no es tan clara como sugieren estas definiciones. Por ejemplo. Un requisito del usuario sobre seguridad podría parecer un requisito no funcional. Sin embargo, cuando se desarrolla en detalle, puede generar otros requisitos que son claramente funcionales, como la necesidad de incluir en el sistema recursos para la autenticación del usuario.

- **Requisitos de usuario**

Los requisitos del usuario para un sistema deben describir los requisitos funcionales y no funcionales de tal forma que sean comprensibles por los usuarios del sistema sin conocimiento técnico detallado. Únicamente deben especificar el comportamiento externo del sistema y deben evitar, tanto como sean posibles las

características del diseño del sistema. Por consiguiente, si se están redactando requisitos del usuario, no se debe utilizar jerga del software, notaciones estructuradas o formales, o describir los requisitos por la descripción de la implementación del sistema. Deben redactarse en un lenguaje sencillo, con tablas y formularios y diagramas intuitivos.

- **Requisitos del sistema**

Los requisitos del sistema son versiones extendidas de los requisitos del usuario que son utilizados por los ingenieros de software como punto de partida para el diseño de sistemas. Agregan detalle y explican cómo el sistema debe proporcionar los requisitos del usuario. Pueden ser utilizados como parte del contrato para la implementación del sistema y, por lo tanto, deben ser una especificación completa y consistente del sistema externo.

En teoría los requisitos del sistema simplemente deben describir el comportamiento externo del sistema y sus restricciones operativas. No deben tratar de cómo se debe diseñar o implementar el sistema. Sin embargo, en el nivel de detalle requerido para especificar completamente un sistema software complejo, es imposible, en la práctica, excluir toda la información del diseño. Existen varias razones para esto:

- Puede tener que diseñar una arquitectura inicial del sistema para ayudar a estructurar la especificación de requisitos.
- En muchos casos, los sistemas deben interoperar con otros ya existentes. Esto restringe el diseño, y estas restricciones imponen requisitos en el sistema nuevo.
- Es necesario el uso de una arquitectura específica para satisfacer los requisitos no funcionales (Como la programación por n versiones para conseguir fiabilidad). Un regulador externo que necesita certificar que el sistema es seguro puede especificar que un diseño arquitectónico que y ha sido certificado sea utilizado.

## 2.4. Ingeniería de Requisitos

La ingeniería de requisitos es definida como:

“[7] Ingeniería de Requisitos ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, qué es lo que el cliente quiere y cómo interactuarán los usuarios finales con el software”.

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

## 2.5. Proceso de la ingeniería de requisitos [8]

La figura 2-2 Proceso de ingeniería de Requerimientos, muestra el proceso en la ingeniería de requisitos

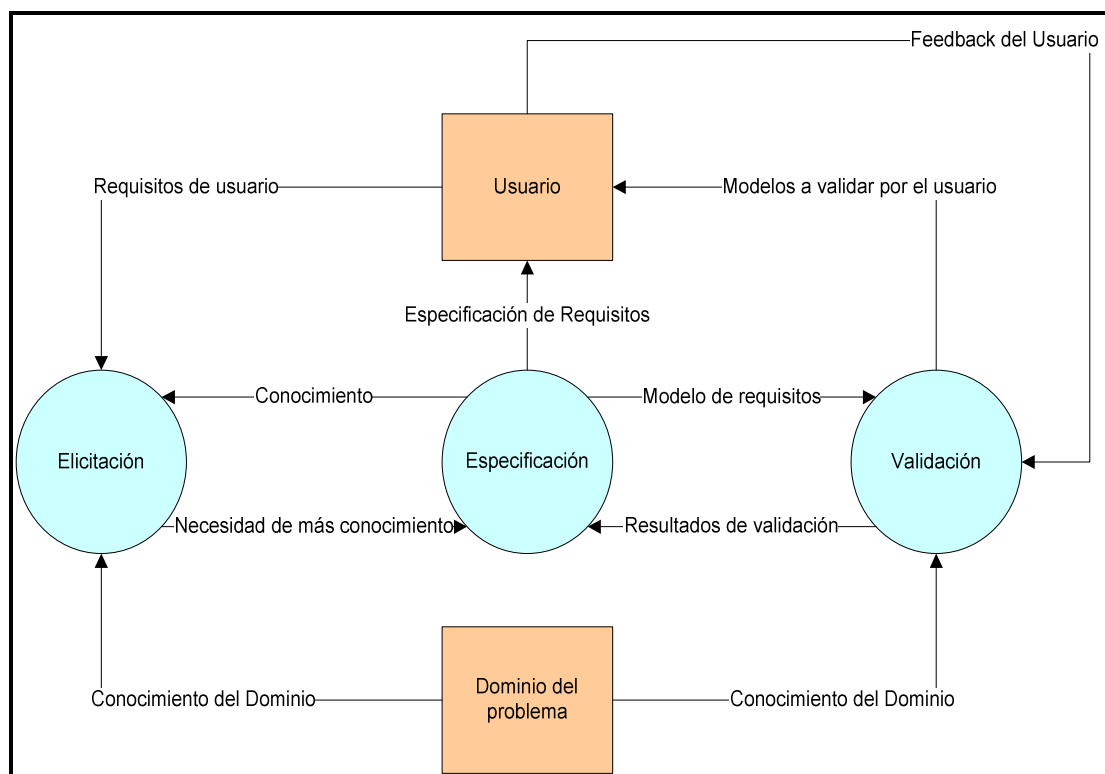


Figura 2-2 Proceso de ingeniería de Requerimientos [8]



- **Elicitación de Requisitos**

En la mayoría de los casos, al comenzar un proyecto de software, el analista conoce muy poco acerca del problema a resolver. La elicitación de requerimientos es el proceso que consiste en adquirir todo el conocimiento relevante, necesario para producir un modelo de requerimientos (especificación) de un dominio de problema. El analista debe realizar la especificación de requerimientos y subsecuentemente su validación con el usuario, solamente después de comprender la naturaleza, características y límites de un problema.

La “captura” de conocimiento es un problema en sí mismo dado que el conocimiento no siempre está disponible en un formato que pueda ser usado por el analista, y además, no es fácil elicitación de conocimiento desde su fuente, especialmente cuando la fuente es un “experto” humano.

Una de las metas más importantes de la elicitación es descubrir cuál es el problema que se debe resolver y, por consiguiente, identificar los límites del sistema. Estos límites definen, a un alto nivel, dónde se adecuará el sistema final entregado en el ambiente operacional actual. La identificación y el acuerdo de los límites de un sistema afectan todas las tareas posteriores a la elicitación.

Otras fuentes de conocimiento del dominio:

- Literatura sobre el dominio
- Software existente en el dominio
- Software similar en otros dominios
- Standards nacionales e internacionales
- Otros “stakeholders” (personas afectadas de un modo u otro por el Sistema a construir) de la organización dónde se implantará en nuevo sistema de software.

Las actividades que abarcan las tareas del analista incluyen la identificación de todas las fuentes de conocimiento de requerimientos, adquirir conocimiento, decidir sobre la relevancia del conocimiento de un problema, y comprender su significado y cómo éste impacta sobre los requerimientos de software.

- **Especificación de Requisitos**

Una especificación puede ser vista como un contrato entre usuarios y desarrolladores de software, el cual define el comportamiento funcional del artefacto de software (y otras propiedades tales como performance, confiabilidad, seguridad, etc.), sin mostrar cómo será obtenido ese comportamiento.

Es importante ver la especificación como un proceso complejo que requiere “feedback” desde el analista al usuario y viceversa. El proceso es analítico debido a que diferentes clases de conocimiento que el analista elicitaba de un dominio de problema, debe ser examinado y posteriormente vinculado de algún modo. También es sintético debido a que conocimiento heterogéneo debe ser combinado para producir una especificación de requerimientos coherente.

Por lo tanto, la especificación de requerimientos puede ser descrita en términos de dos actividades principales:

- Análisis y asimilación de conocimiento de requerimientos
- Síntesis y organización de conocimiento en un modelo de requerimientos lógico y coherente

Se considera que la especificación de requerimientos produce una variedad de modelos, los cuales corresponden a diferentes visiones del problema. Es así como la especificación de requerimientos produce:

- Modelos orientados al usuario, especificando el comportamiento y características no funcionales del software que servirá como punto de entendimiento entre el analista, el cliente y el usuario.
- Modelos orientados al desarrollador, especificando propiedades funcionales y no funcionales del sistema de software, así como restricciones sobre recursos, restricciones sobre diseño, etc. Estos modelos son importantes de considerar en etapas de desarrollo posteriores.

Esta diferencia de modelos es muy útil, ya que notaciones de especificación claras para desarrolladores pueden ser difíciles de comprender por los usuarios.

La especificación de requerimientos es el proceso central de ingeniería de requerimientos. Actúa como medio de control para los procesos de elicitación y de validación.

Durante la especificación puede surgir la necesidad de mayor información acerca del problema. Esta necesidad dispara nuevamente el proceso de elicitación en búsqueda de información. Por otra parte, algún cambio en el dominio del problema debe producir cambios en la especificación. De este modo, puede requerirse elicitación durante la especificación.

Similar situación ocurre en la interacción con la validación, proceso que se describe a continuación.

- **Validación de requisitos [9]**

La validación de requisitos trata de mostrar que estos realmente definen el sistema que el cliente desea. Coincide parcialmente con el análisis ya que este implica encontrar problemas con los requisitos. La validación del requisito es importante debido a que los errores en el documento de requisitos pueden conducir a importantes costes al repetir el trabajo cuando estos son descubiertos durante el desarrollo o después que el sistema este en uso. El coste de arreglar un problema en los requisitos haciendo un cambio en el sistema es mucho mayor que reparar los errores de diseño o los de codificación. La razón de esto es que un cambio en los requisitos normalmente significa que el diseño y la implementación del sistema también deben cambiar y que este debe probarse nuevamente.

Durante el proceso de validación de requisitos, se deben llevar a cabo verificaciones sobre requisitos en el documento de requisitos. Estas verificaciones comprenden:

- **Verificaciones de validez.** Un usuario puede pensar que se necesita un sistema para llevar a cabo ciertas funciones. Sin embargo, el razonamiento y el análisis pueden identificar que se requieren funciones adicionales o diferentes. Los sistemas tienen diversos stakeholders con diferentes necesidades, y cualquier

conjunto de requerimientos es inevitablemente un compromiso en el entorno del stakeholder.

- **Verificaciones de consistencia.** Los requerimientos en el documento no deben contradecirse. Esto es, no debe haber restricciones o descripciones contradictorias de la misma función del sistema.
- **Verificaciones de completitud.** El documento de requerimientos debe incluir requerimientos que definan todas las funciones y restricciones propuestas por el usuario del sistema.
- **Verificaciones de realismo.** Utilizando el conocimiento de la tecnología existente, los requerimientos deben verificarse para asegurar que se pueden implementar. Estas verificaciones también deben tener en cuenta el presupuesto y la confección de agendas para el desarrollo del sistema.
- **Verificabilidad.** Para reducir la posibilidad de discusiones entre el cliente y el contratista, los requerimientos del sistema siempre deben redactarse de tal forma que sean verificables. Esto significa que debe poder escribir un conjunto de pruebas que demuestren que el sistema a entregar cumple cada uno de los requerimientos especificados.

## 2.6. Actualidad en la Elicitación de Requisitos

- **Comparación del Proceso de Elicitación de Requerimientos en el desarrollo de Software a Medida y Empaquetado. Propuesta de métricas para la Elicitación [10]**

El trabajo de tesis tiene como objetivo la comparación de las técnicas de elicitación de requerimientos aplicables a la ingeniería de software ya sean éstas tanto para el desarrollo de software a medida como empaquetado. El análisis se realizó en base a dos encuestas, una destinada a desarrolladores de software y

otra a usuarios considerando una serie de buenas prácticas recomendadas para el proceso de elicitación de requerimientos. Los resultados permitieron comprobar la hipótesis que establece que la alternativa de software a desarrollar (a medida o empaquetado) condiciona el proceso de elicitación.

- **Aplicación de la Teoría de Constructos Personales a la Elicitación de Requisitos [11]**

En este artículo se exploran varias aplicaciones de una teoría psicológica, la Teoría de Constructos Personales para la elicitación de objetivos. Esta teoría tiene la ventaja de tener una base estadística y, por tanto, ser más rigurosa e independiente, y la posibilidad de ser automatizada utilizando la técnica de la rejilla.

- **Alineación entre metas organizacionales y elicitación de requisitos del Software [12]**

La elicitación de los requisitos del software debe contribuir a la obtención de una primera descripción del problema que permita establecer las motivaciones para su solución mediante sistemas informáticos. Los métodos y herramientas convencionales para la elicitación de requisitos pueden, sin embargo, dejar de lado aspectos fundamentales de la organización debido a su interés por enfocarse en situaciones relativas a los problemas puntuales que aquejan el entorno del software. En este artículo se ilustra el uso de una herramienta administrativa de uso común (el diagrama causa - efecto), con algunas modificaciones, para lograr un análisis cuantitativo del grado en que se logran alinear las metas organizacionales con los requisitos correspondientes a una pieza de software.

- **Elicitación de Objetivos a partir de Escenarios [13]**

La Ingeniería de Requisitos se ocupa de los aspectos de la Ingeniería de Software, relacionados con la comprensión y producción de descripciones de problemas para resolverlos a través de la construcción de Sistemas de Software. Los fundamentos del sistema (el por qué) están abarcados por los “objetivos” de la organización, y se definen usualmente como las metas a ser cumplidas por el sistema y su entorno. Como todo producto de la fase de requerimientos, los objetivos del sistema deben recorrer un proceso de elicitación, modelización y validación. Sin embargo, los enfoques orientados por objetivos, si bien poseen, aunque en diferente medida, procesos mediante los que modelizan y validan los objetivos, no sucede lo mismo en cuanto a las indicaciones para la elicitación de los objetivos. Este trabajo presenta la utilización de Escenarios como fuente para la elicitación de Objetivos, para lo que se propone una estrategia de identificación de Objetivos del modelo GBRAM de Antón a partir del esquema de Escenarios de Leite.

- **Elicitación de requisitos en sistemas de gestión orientados a procesos [14]**

En el artículo se propone utilizar modelos de gestión basada en procesos como guía para elicitar requisitos del futuro sistema. La aproximación permite relacionar aspectos de alto nivel de la organización (metas estratégicas) con requisitos del sistema

- **Una metodología para elicitación de requisitos en proyectos GSD [15]**

La cantidad de proyectos de desarrollo global de software aumenta día a día, y sus beneficios se ven desafiados por los problemas que imponen las grandes distancias entre los sitios de desarrollo y la necesidad de comunicarse con personas con las que generalmente no se comparte el lenguaje ni la cultura. Durante el proceso de elicitación de requisitos, estas dificultades suelen aumentar, debido al rol crucial que juega la comunicación en este proceso. En

este artículo presentamos una metodología desarrollada para mejorar el proceso de elicitación global de requisitos, así como el diseño del experimento que se llevará a cabo próximamente para validarla.

## 2.7. Herramientas para la Gestión de Requisitos en la actualidad

- **SIGERAR: Una herramienta para la Gestión de Requisitos [16]**

En la actualidad, el uso de la ingeniería Requisitos es una de las maneras más seguras lograr la calidad en el desarrollo y mantenimiento de sistemas de software, ya que esta práctica reduce errores apreciables, los fracasos y las ambigüedades de producto final que se entregará. Con el tiempo, se producen cambios en los requisitos, debido a diversos factores tales como errores, inconsistencias, los problemas desarrollo organizacional del conocimiento del stakeholders<sup>1</sup>, los cambios legales, etc., lo que requiere una gran esfuerzo de las empresas para controlar y gestión de los mismos. En este artículo es proporcionar una herramienta automatizada para la gestión de requisitos, llamado SIGER. La herramienta recopila, almacena y mantiene los requisitos, gestión del cambio y la promoción de trazabilidad entre los requisitos y entre requisitos de los documentos. La contribución de este trabajo es proporcionar a los desarrolladores de software una herramienta para los requisitos de gestión, el uso Libre y fácil instalación, configuración y operación, adherente al ciclo de vida completo de software, proporcionando un control y análisis de riesgo, el impacto y los costos de los cambios en los requisitos.

- **GERESOFT – Gestión de la Verificación y Validación de Requisitos[17]**

En este artículo se presenta la herramienta GERESOFT, como un Software para la Verificación y Validación de la Gestión de Requisitos, tomando como base para su diseño y construcción lineamientos del Software Libre, estándares de la IEEE y conceptos relacionados con la V&V(Verificación y Validación). Para

realizar el proceso de V&V se utilizaron las Revisiones como Técnica de V&V bajo la orientación del Análisis Estático, y desde el punto de vista de la Ingeniería de Software como insumo para aplicar el conocimiento generado con este desarrollo se utilizó la Ingeniería de Requisitos, la cual es el proceso fundamental que permite planificar, administrar y desarrollar

- **Mejora de Calidad en Ingeniería de Requisitos Mediante la Aplicación de Metodología Six-Sigma. [18]**

Six-Sigma reconoce que existe una correlación directa entre el número de productos defectuosos, las pérdidas operacionales y el nivel de satisfacción de los clientes. En este artículo, se propone la aplicación de Six-Sigma al proceso de Ingeniería de Requisitos (IR), centrado en el mejoramiento de la calidad mediante la comprensión y la reducción de la variación del proceso de educación de requisitos, con lo cual se espera alcanzar un proceso de IR, con un nivel de calidad que se aproxime al ideal de cero-defectos. Para la aplicación de Six-Sigma al proceso de Ingeniería de Requisitos, sus agentes de cambio, deberán adaptarse y dimensionarse de acuerdo a los perfiles de los participantes que intervienen en el desarrollo de productos de software. Además, se requerirá del uso intensivo de herramientas estadísticas para eliminar la variabilidad de los requisitos y producir los resultados esperados, con el mínimo posible de defectos, bajos costos y máxima satisfacción del cliente aplicando técnicas de IR en conjunto con las de calidad tradicional y Gestión de Calidad Total (TQM).

## **2.8. Modelos de Calidad**

### **2.8.1 Norma Técnica peruana ISO 12207 Procesos del Ciclo de Vida del Software [19]**

La Norma técnica Peruana ISO 12207, define el proceso de la gestión de requisitos de la siguiente forma:



**Análisis de los requerimientos del sistema:** Esta actividad consta de las siguientes tareas, que el desarrollador deberá llevar a cabo o proporcionar apoyo, según requiera el contrato:

- Se deberá analizar el uso específico previsto del sistema a ser desarrollado para especificar los requerimientos del sistema. La especificación de los requerimientos del sistema deberá describir funciones y capacidades del sistema; requerimientos de negocio, organizativos y de usuario; requerimientos de seguridad física y de acceso; requerimientos de ingeniería de factores humanos (ergonomía), interfaces y requerimientos de operación y mantenimiento; limitaciones de diseño y requerimientos de calificación. Se deberá documentar la especificación de los requerimientos del sistema.
  
- Se deberán evaluar los requerimientos del sistema teniendo en cuenta los criterios enumerados a continuación. Se deberán documentar los resultados de las evaluaciones.
  - a) Trazabilidad hacia las necesidades de la adquisición.
  - b) Consistencia con las necesidades de la adquisición.
  - c) Capacidad para ser probados.
  - d) Viabilidad del diseño de la arquitectura del sistema.
  - e) Viabilidad de la operación y mantenimiento.

**Análisis de los requerimientos software:**

Para cada elemento software (o para cada elemento de configuración software, si se ha identificado) esta actividad consta de las siguientes tareas:

- El desarrollador deberá establecer y documentar los requerimientos software descritos a continuación, incluyendo la especificación de las

características de calidad. Se pueden encontrar guías para la especificación de las características de calidad en la NTPISO/ IEC 9126.

- a) Especificaciones funcionales y de capacidad, incluyendo prestaciones, características físicas y condiciones del entorno en donde el elemento software ha de funcionar.
  - b) Interfaces externas al elemento software.
  - c) Requerimientos de calificación.
  - d) Especificaciones de seguridad física, incluyendo aquellas relacionadas con los métodos de operación y mantenimiento, influencias del entorno y daño a las personas.
  - e) Especificaciones de seguridad de acceso, incluyendo aquellas que comprometen información confidencial.
  - f) Especificaciones relacionadas con ingeniería de factores humanos (ergonomía), incluyendo aquellas relacionadas con las operaciones manuales, interacción hombre-máquina, obligaciones del personal y áreas con necesidad de una especial atención por parte de las personas, debido a su sensibilidad a errores humanos y a la destreza.
  - g) Definición de datos y requerimientos de las bases de datos.
  - h) Requerimientos de instalación y aceptación del producto software entregado, en el lugar o lugares de operación y mantenimiento.
  - i) Documentación de usuario.
  - j) Requerimientos de operación y ejecución por parte del usuario.
  - k) Requerimientos de mantenimiento por parte del usuario.
- El desarrollador deberá evaluar los requerimientos software teniendo en cuenta los criterios enumerados a continuación. Se deberán documentar los resultados de la evaluación.
    - a) Trazabilidad hacia los requerimientos del sistema y el diseño del sistema.

- b) Consistencia externa con los requerimientos del sistema.
  - c) Consistencia interna.
  - d) Capacidad para ser probado.
  - e) Viabilidad del diseño software.
  - f) Viabilidad de la operación y mantenimiento.
- El desarrollador deberá llevar a cabo revisiones conjuntas de acuerdo con el apartado 6.6.

### 2.8.2 Modelo de Calidad CMMI [20]

CMMI nivel 2 pretende conseguir que en los proyectos de la organización haya una gestión de los requisitos y que los procesos estén planeados, ejecutados, medidos y controlados.

La tabla 2-1 Vinculación de las etapas del proceso con Las prácticas específicas de CMMI, muestra la vinculación de las etapas del proceso de la gestión de requisitos con las prácticas específicas del CMMI.

**Tabla 2-1 Vinculación de las etapas del proceso con  
Las practicas específicas de CMMI**

Gestión de Requisitos	CMMI practicas específicas
Definición de Requisitos	SP1.1 Obtención y entendimiento de los requisitos
Análisis y Negociación	SP1.2 Obtener el acuerdo sobre los requisitos (criterios de aceptación)
Documentación	SP1.1 Obtención y entendimiento de los requisitos
Validación	SP1.2 Obtener el acuerdo sobre los requisitos, SP1.5 Identificar inconsistencias entre los productos de trabajo del proyecto y los requisitos

Gestión de los cambios	SP1.3 Gestión de los cambios de requisitos, SP1.4 Mantenerla trazabilidad bidireccional de los requisitos, SP1.5 Identificar inconsistencias entre los productos de trabajo del proyecto y los requisitos
------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

- **SG 1 Gestionar requisitos**

*Los requisitos son gestionados y las inconsistencias con el plan de proyecto y entregables son identificados*

El proyecto mantiene un conjunto de requisitos aprobados durante la vida del proyecto de la siguiente manera:

- Gestionando todos los cambios a los requisitos
- Manteniendo las relaciones entre los requisitos, los planes del proyecto y sus entregables.
- Identificando inconsistencias entre los requisitos, los planes del proyecto y sus entregables.
- Tomando acciones correctivas.

- **SP 1.1 Lograr un entendimiento de los requisitos**

*Desarrollar un entendimiento con quienes proporcionan requisitos acerca del significado de los requisitos*

Según el proyecto madura, y se van derivando requisitos, todas las actividades o disciplinas recibirán requisitos. Para evitar requisitos ocultos, se establecen criterios para designar los canales apropiados o fuentes oficiales desde las cuales se deben recibir los requisitos. Las actividades de recepción de requisitos llevan al análisis de los requisitos con sus proveedores para asegurar que son compatibles, que existe un entendimiento común sobre el significado de los

mismos. El resultado de este análisis y diálogo es un conjunto de requisitos acordados.

### **Entregables típicos**

- Lista de criterios para distinguir apropiadamente a los proveedores de requisitos.
- Criterios para la evaluación y aceptación de requisitos.
- Resultados de análisis de criterios
- Un conjunto acordado de requisitos

### **Subprácticas**

- Establecer criterios para distinguir apropiadamente a los proveedores de requisitos.
- Establecer criterios objetivos para la evaluación y aceptación de requisitos.

*La falta de evaluación y criterios de aceptación frecuentemente resultan en una verificación no adecuada, con costos de re-trabajo o rechazo del cliente.*

*Ejemplos de evaluación de criterios de aceptación incluyen los siguientes:*

- *Clara y apropiadamente indicados*
- *Completos*
- *Consistentes unos con otros*
- *Identificados de modo único*
- *Verificables (que se pueden probar)*
- *Trazables*

- Analizar requisitos para asegurar que se satisfagan los criterios establecidos.

- Conseguir un entendimiento de los requisitos con el proveedor de los mismos y así conseguir que los participantes del proyecto puedan comprometerse con ellos.

- **SP 1.2 Obtener compromiso con los requisitos**

***Obtener compromiso con los requisitos de parte de los participantes del proyecto***

Mientras la anterior práctica específica conseguía una comprensión con los proveedores de los requisitos, esta práctica específica busca obtener el acuerdo y el compromiso entre aquellos que han de realizar las actividades necesarias para implementar los requisitos.

Los requisitos evolucionan durante todo el proyecto, especialmente en la manera en que se describen en las prácticas específicas de las áreas de proceso Desarrollo de requisitos y Solución técnica. Como los requisitos evolucionan, esta práctica específica asegura que los participantes en el proyecto se comprometan con los últimos requisitos aprobados, considerando los cambios realizados en los planes de proyecto, actividades y entregables.

**Entregables típicos**

- Evaluación de impacto de los requisitos.
- Compromisos documentados respecto a los requisitos y a sus cambios.

**Subprácticas**

- Evaluar el impacto de los requisitos sobre compromisos existentes. Cuando los requisitos cambian o se agrega un nuevo requisito se debería evaluar el impacto sobre los participantes del proyecto.
- Negociar y registrar los compromisos. Antes que los participantes del proyecto se comprometan con los requisitos o cambios a los mismos, se deben negociar los cambios a los compromisos existentes.

- **SP 1.3 Gestionar cambios a los requisitos**

***Gestionar cambios a los requisitos conforme evolucionan durante el proyecto***

Durante el proyecto, los requisitos cambian por varias razones. Tanto por cambios en las necesidades como en los procedimientos de trabajo, se derivan requisitos adicionales que podrían afectar a los requisitos existentes. Es esencial gestionar estas incorporaciones y cambios de modo eficiente y efectivo. Para analizar de modo efectivo el impacto de los cambios, es necesario que sea conocido el origen de cada requisito, y la razón por cada cambio que se documente. Sin embargo el gerente de proyecto puede monitorear métricas apropiadas de la volatilidad de los requisitos para juzgar si se necesita revisar o implementar nuevos mecanismos de control.

**Entregables típicos**

- Estado de los requisitos
- Base de datos de requisitos
- Base de datos de decisiones relacionadas a requisitos

**Subprácticas**

- Documentar todos los requisitos y cambios a los requisitos que han sido datos o generados por el proyecto.
- Mantener el historial de cambios a los requisitos con el motivo de los cambios. Mantener el historial de cambios ayuda a monitorear la volatilidad de los requisitos.
- Evaluar el impacto de los cambios a los requisitos desde el punto de vista de los stakeholders relevantes.
- Garantizar la disponibilidad de los requisitos y la información de sus cambios al proyecto.

- **SP 1.4 Mantener trazabilidad bidireccional de los requisitos**

***Mantener trazabilidad bidireccional entre los requisitos y los planes de proyecto y entregables***

La intención de esta práctica específica es mantener la trazabilidad bidireccional de los requisitos por cada nivel de descomposición del producto. (Ver la definición de “trazabilidad bidireccional” en el glosario). Cuando los requisitos se gestionan adecuadamente, la trazabilidad puede ser establecida desde el origen del requisito hacia el requisito de más bajo nivel y desde el requisito de más bajo nivel hacia su origen. Por otro lado, la trazabilidad bidireccional ayuda a determinar que todos los orígenes de los requisitos se han tomado en cuenta y que todos los requisitos de bajo nivel pueden ser trazados hacia un origen válido.

La trazabilidad de requisitos puede también cubrir las relaciones hacia otras entidades, tales como los entregables intermedios y finales, cambios en la documentación de diseño y planes de prueba. La trazabilidad puede cubrir relaciones horizontales, tales como las que se hacen a través de interfaces, así como también relaciones verticales. La trazabilidad es particularmente necesaria cuando se realiza una evaluación del impacto de un cambio a los requisitos sobre las actividades del proyecto y entregables.

**Entregables típicos**

- Matriz de trazabilidad de requisitos.
- Sistema de monitoreo de requisitos.

**Subprácticas**

- Mantener la trazabilidad de requisitos para asegurar que el origen de los requisitos de bajo nivel (requisitos derivados) se encuentre documentado.



- Mantener la trazabilidad de requisitos desde un requisito a su requisito derivado y asignarlos a funciones, interfaces, objetos, personas, procesos y entregables.
  - Generar la matriz de trazabilidad de requisitos.
- **SP 1.5 Identificar inconsistencias entre entregables y requisitos.**

***Identificar inconsistencias entre los requisitos y los planes y entregables del proyecto***

Esta práctica específica encuentra las inconsistencias entre los requisitos y los planes del proyecto y los entregables e inicia las acciones correctivas para corregirlos.

**Entregables típicos**

- Documentación de inconsistencias, incluyendo orígenes, condiciones y razones.
- Acciones correctivas

**Subprácticas**

- Revisar los planes del proyecto, actividades y entregables para conseguir la consistencia con los requisitos y los cambios realizados a ellos.
- Identificar la fuente de la inconsistencia y la razón.
- Identificar los cambios que se necesiten a los planes y a los entregables que resulten de los cambios a la línea base de los requisitos.
- Iniciar las acciones correctivas.

## 2.9. Métricas

### 2.10.1 METRICA 3 [26]

MÉTRICA Versión 3 tiene un enfoque orientado al proceso, ya que la tendencia general en los estándares se encamina en este sentido y por ello se ha enmarcado dentro de la norma ISO 12 207, que se centra en la clasificación y definición de los procesos del ciclo de vida del software.

- **ESTABLECIMIENTO DE REQUISITOS**

En esta actividad se lleva a cabo la definición, análisis y validación de los requisitos a partir de la información facilitada por el usuario, completándose el catálogo de requisitos obtenido en la actividad Definición del Sistema (ASI 1). El objetivo de esta actividad es obtener un catálogo detallado de los requisitos, a partir del cual se pueda comprobar que los productos generados en las actividades de modelización se ajustan a los requisitos de usuario.

Esta actividad se descompone en un conjunto de tareas Tabla 2-2 Tareas del establecimiento de requisitos, que si bien tienen un orden, exige continuas realimentaciones y solapamientos, entre sí y con otras tareas realizadas en paralelo. No es necesaria la finalización de una tarea para el comienzo de la siguiente. Lo que se tiene en un momento determinado es un catálogo de requisitos especificado en función de la progresión del proceso de análisis.

Se propone como técnica de obtención de requisitos la especificación de los casos de uso de la orientación a objetos, siendo opcional en el caso estructurado. Dicha técnica ofrece un diagrama simple y una guía de especificación en las sesiones de trabajo con el usuario.

Tabla 2-2 Tareas del establecimiento de requisitos

	<b>Tarea</b>	<b>Productos</b>	<b>Técnicas y Prácticas</b>	<b>Participantes</b>
<b>ASI 2.1</b>	Obtención de Requisitos	<ul style="list-style-type: none"> <li>• Catálogo de Requisitos</li> <li>• Modelo de Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Sesiones de Trabajo</li> <li>• Catalogación</li> <li>• Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Usuarios Expertos</li> <li>• Analistas</li> </ul>
<b>ASI 2.2</b>	Especificación de Casos de Uso	<ul style="list-style-type: none"> <li>• Catálogo de Requisitos</li> <li>• Modelo de Casos de Uso</li> <li>• Especificación de Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Sesiones de Trabajo</li> <li>• Catalogación</li> <li>• Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Usuarios Expertos</li> <li>• Analistas</li> </ul>
<b>ASI 2.3</b>	Análisis de Requisitos	<ul style="list-style-type: none"> <li>• Catálogo de Requisitos</li> <li>• Modelo de Casos de Uso</li> <li>• Especificación de Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Sesiones de Trabajo</li> <li>• Catalogación</li> <li>• Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Usuarios Expertos</li> <li>• Analistas</li> </ul>
<b>ASI 2.4</b>	Validación de Requisitos	<ul style="list-style-type: none"> <li>• Catálogo de Requisitos</li> <li>• Modelo de Casos de Uso</li> <li>• Especificación de Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Sesiones de Trabajo</li> <li>• Catalogación</li> <li>• Casos de Uso</li> </ul>	<ul style="list-style-type: none"> <li>• Usuarios Expertos</li> <li>• Analistas</li> </ul>

• **Tarea ASI 2.1: Obtención de Requisitos**

En esta tarea comienza la obtención detallada de información mediante sesiones de trabajo con los usuarios, previamente identificados en la actividad Definición del Sistema (ASI 1).

Se recoge información de los requisitos que debe cumplir el software. En la definición de los requisitos, que sirven de base para establecer los niveles de servicios del sistema, hay que tener en cuenta, si existen, las posibles restricciones del entorno, tanto hardware como software, que puedan afectar al sistema de información.

También se definen las prioridades que hay que asignar a los requisitos, considerando los criterios de los usuarios acerca de las funcionalidades a cubrir.

Los principales tipos de requisitos que se deben especificar son, por ejemplo:

- Funcionales.
- Rendimiento.
- Seguridad.
- Implantación.
- Disponibilidad del sistema.

En el caso de orientación a objetos se especifican, además, los casos de uso asociados a los requisitos funcionales.

Los casos de uso son una técnica de especificación de requisitos válida tanto en desarrollos estructurados como en orientación a objetos, aunque en este último caso se propone como técnica obligatoria al ser necesaria como referencia a lo largo de todo el ciclo de vida. En esta tarea se elabora el modelo de casos de uso, según las normas y estándares de la organización, identificando:

- Actores.
- Casos de uso.
- Breve descripción de cada caso de uso.

Los productos obtenidos en la tarea Determinación del Alcance del Sistema (ASI 1.1), son tomados como referencia durante la obtención de requisitos, de forma que todos los requisitos especificados se encuentren dentro del ámbito del sistema de información.

- ***Tarea ASI 2.2: Especificación de Casos de Uso***

Esta tarea es obligatoria en el caso de orientación a objetos, y opcional en el caso de análisis estructurado, como apoyo a la obtención de requisitos.

El objetivo de esta tarea es especificar cada caso de uso identificado en la tarea anterior, desarrollando el escenario.

Para completar los casos de uso, es preciso especificar información relativa a:

- Descripción del escenario, es decir, cómo un actor interactúa con el sistema, y cual es la respuesta obtenida.
- Precondiciones y post condiciones.
- Identificación de interfaces de usuario.

- Condiciones de fallo que afectan al escenario, así como la respuesta del sistema (Escenarios secundarios).

En escenarios complejos, es posible utilizar como técnica de especificación los diagramas de transición de estados, así como la división en casos de uso más simples, actualizando el modelo de casos de uso.

Para la obtención de esta información es imprescindible la participación activa de los usuarios.

- ***Tarea ASI 2.3: Análisis de Requisitos***

En esta tarea se estudia la información capturada previamente en esta actividad, para detectar inconsistencias, ambigüedades, duplicidad o escasez de información, etc.

También se analizan las prioridades establecidas por el usuario y se asocian los requisitos relacionados entre sí.

El análisis de los requisitos y de los casos de uso asociados permite identificar funcionalidades o comportamientos comunes, reestructurando la información de los casos de uso a través de las generalizaciones y relaciones entre ellos.

Mediante sesiones de trabajo con los usuarios, se contrastan las conclusiones del análisis de la información recogida.

- ***Tarea ASI 2.4: Validación de Requisitos***

Mediante esta tarea, los usuarios confirman que los requisitos especificados en el catálogo de requisitos, así como los casos de uso, son válidos, consistentes y completos.

## 2.10. Estudios actuales en la mejorar la calidad en la Gestión de Requisitos

➤ **Mejora de la calidad de los requisitos mediante la generación de pruebas [21]**

Una de las piezas básicas para desarrollar sistemas software en el tiempo planificado y con la calidad adecuada es contar con buenos requisitos. Estos deben ser claros, precisos, no ambiguos y completos. Sin embargo, conseguir buenos requisitos es difícil. Para verificar y mejorar la calidad de los requisitos existen varias técnicas como revisiones formales, matrices de rastreabilidad ontologías, etc. Otra técnica poco utilizada es la generación de pruebas a partir de los requisitos. Este trabajo describe esta técnica y muestra un caso práctico aplicado sobre un sistema real actualmente en explotación.

➤ **Explicitar Requisitos del Software usando Escenarios [22]**

Los escenarios describen situaciones del proceso del negocio, tanto del proceso observable actual como del proceso proyectado o futuro. En este último caso los escenarios resultan ser contenedores de la mayoría de los requisitos del sistema de software, pero no son los requisitos propiamente dichos. Muchas de las buenas prácticas o prácticas recomendadas en el proceso de desarrollo de software se basan en la existencia de un documento de requisitos en el que éstos se individualizan en forma precisa. En el presente artículo se presenta una estrategia para confeccionar un documento de requisitos a partir de escenarios ya construidos. La particularidad que presenta la misma es que los requisitos individualizados tienden a ser libres de conflictos y de ambigüedad a consecuencia del propio proceso de producción.

➤ **Una propuesta para mejorar la completitud de requisitos utilizando un enfoque lingüístico[23]**

En el artículo se presenta una propuesta para mejorar la calidad, a nivel de completitud, de especificaciones de requisitos escritas en *español restringido*.

La propuesta está basada en la *gramática de casos* planteada por Fillmore y en un diálogo con el usuario que permite adquirir la información ausente detectada en la especificación.

➤ **Estimación de Completitud en Modelos de Requisitos Basados en Lenguaje Natural [24]**

En el presente artículo se estudia experimentalmente el uso de los datos de captura y recaptura en el proceso de la Ingeniería de Requisitos, en particular en la construcción de Escenarios a partir del Léxico Extendido del Lenguaje. Se analiza también el impacto que la completitud del LEL tiene sobre la completitud del modelo de escenarios correspondiente.

➤ **Generación de Especificaciones de Requisitos de Software a partir de Modelos de Negocios: un enfoque basado en metas[25]**

Este enfoque permitirá asegurar que el sistema de información a desarrollar sea realmente útil en las tareas de los actores organizacionales. Los trabajos de investigación en esta área han determinado que las metas organizacionales son una buena base para establecer la relación entre los objetivos perseguidos por el negocio y los requisitos del sistema de información a desarrollar, ya que todos estos requisitos (funcionales y no funcionales) deben corresponderse con tareas que se desean desempeñar dentro de un proceso de negocios. Los procesos de negocio a su vez, permiten el cumplimiento o satisfacción de alguna o algunas de las metas del negocio. En este trabajo se presenta una propuesta para la obtención de requisitos de software a partir de modelos de negocios.

### 3. CAPITULO III: DEFINICIÓN DEL MÉTODO DE ELECCIÓN DEL MODELO DE CALIDAD PARA MEJORAR LA COMPLETITUD DE REQUISITOS EN LOS COLEGIOS PRIVADOS DE LIMA

El método de selección se basa en una mezcla de investigaciones que han sido realizadas en el campo de SPI (Software Process Improvement) [27][28][29][30][31][32].

Del estudio [27] se ha tomado como referencia la división del método en 2 etapas bien diferenciadas, y la forma en que se ejecuta la primera de las etapas que involucra la obtención de la información y el análisis de la misma obteniendo como resultado los factores. De los estudios [28][29][30][31] y [32] se obtuvo la manera en la que definen la escala de medición y la tabulación de los Modelos contra los factores.

El método de selección elaborado esta dividido principalmente en 2 etapas

**Primera Etapa:** Analizar el proceso de Gestión de Requisitos

La primera etapa es dividida en las siguientes actividades:

- i. Identificación de las Necesidades en la Gestión de Requisitos
- ii. Identificación de los Factores

**Segunda Etapa:** Elección de un Modelo de Calidad

La segunda etapa es dividida en las siguientes actividades:

- iii. Selección de los Criterios
- iv. Elección del Modelo de Calidad



### **3.1. Planteamiento de la Solución**

Se plantea la elección de un Modelo de calidad, la primera etapa incluye la identificación de las necesidades presentes en el proceso de la Gestión de Requisitos. Las necesidades listadas son las entradas que nos permitirán identificar los principales factores que afectan la Gestión de Requisitos que se hallaran en base al método Scoring.

En la segunda etapa se definirán los criterios basado en la identificación de las similitudes entre modelos de calidad elegidos, estos criterios se utilizarán en la elección del Modelo de calidad más adecuado para los colegios privados de Lima. Seguidamente se procederá con la tabulación de los modelos de Calidad contra los factores, y en base a los criterios identificados se colocará un nivel de satisfacción de cada factor utilizando la escala Likert.

Al terminar la tabulación se procederá con la sumatoria de los puntajes, y se determinará cual es el modelo que satisface más cada factor según los criterios, con estos resultados se realizara una interpretación y la elaboración de conclusiones.

### **3.2. Definición del método de elección del Modelo de Calidad**

#### **3.2.1 Primera Etapa: Analizar el proceso de Gestión de Requisitos**

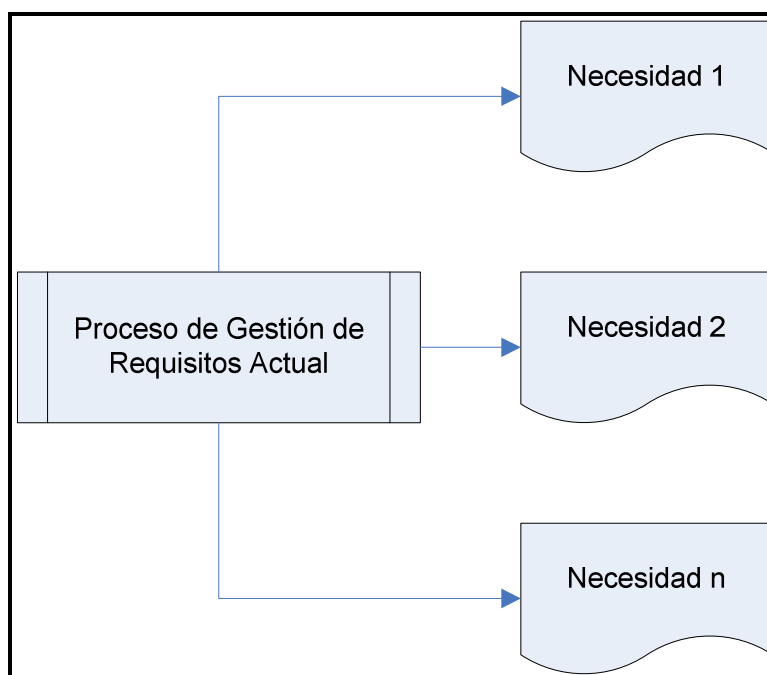
##### **3.2.1.1 Identificar necesidades en la Gestión de Requisitos**

Para la identificación de las necesidades se plantea primeramente la recopilación de toda la información posible, tales como:

- ❖ El Diagrama del procesos de desarrollo de aplicaciones informáticas, centrándonos en el tema de interés, la Gestión de Requisitos
- ❖ Audio de entrevistas acerca de cómo se Gestiona actualmente los requisitos, etc.

Seguidamente se procederá con la revisión de toda la información recopilada y a partir de ello se identificarán las necesidades en la Gestión de Requisitos generando como salida un listado de necesidades.

La figura 3-1 Identificación de necesidades en la Gestión de Requisitos, muestra como un rectángulo el proceso de la Gestión de Requisitos, de donde se identificarán las necesidades que son representadas en la figura como documentos.



**Figura 3-1 Identificación de Necesidades en la Gestión de Requisitos**  
(Elaboración Propia)

### 3.2.1.2 Identificación de factores

En la identificación de los factores que afectan la Gestión de Requisitos en los colegios privados de Lima, se procederá listando las necesidades identificadas previamente y seleccionando las principales en base al método de Scoring, obteniendo como resultado el listado de factores que afectan la Gestión de Requisitos. En figura 3-2 observamos gráficamente las necesidades que son entradas para el Método Scoring (Ver Anexo II) generando el listado de factores.

El Método Scoring consiste:

Etapas:

- ❖ Identificación de la meta general del problema
- ❖ Identificación de alternativas
- ❖ Listar los criterios a emplear en la toma de decisiones
- ❖ Asignar una ponderación para cada uno de los criterios
- ❖ Establecer en cuanto satisface cada alternativa a nivel de cada uno de los criterios
- ❖ Calcular la puntuación de cada una de las alternativas

Se utilizará la escala Likert de 5 puntos para la evaluación de cada factor, en donde se distribuyen los puntajes de la siguiente manera:

1: Muy poco importante

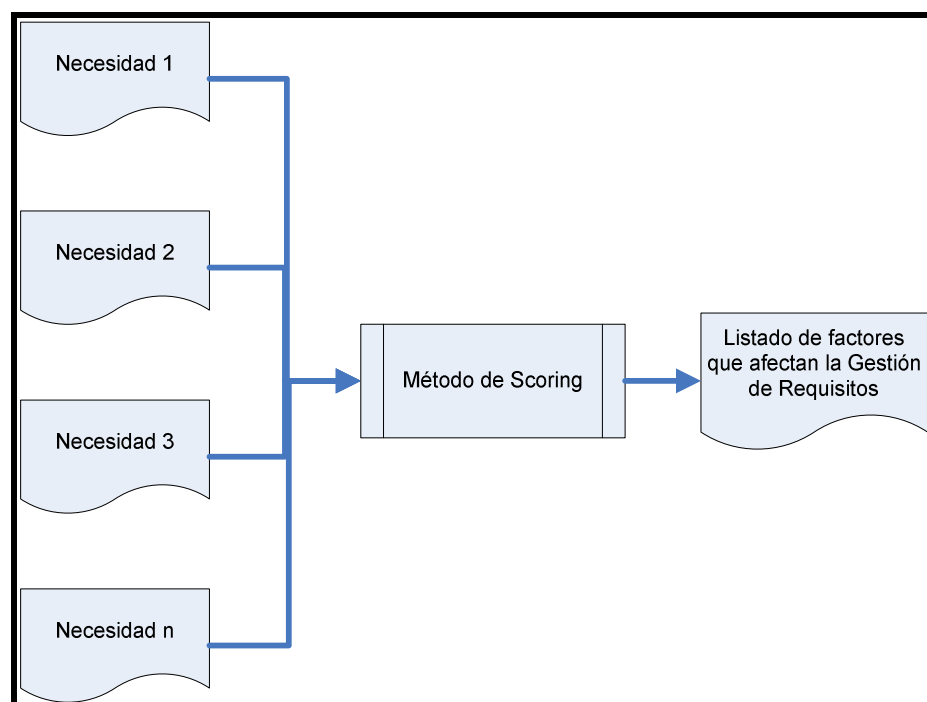
2: Poco importante

3: Importancia media

4: Algo Importante

5: Muy importante

- ❖ Ordenar las alternativas en función de la puntuación



**Figura 3-2 Factores que afectan la gestión de Requisitos**

### **3.2.2 Segunda Etapa: Elección de un Modelo de Calidad**

#### **3.2.2.1 Selección de criterios**

Para la elección de los criterios se procederá de la siguiente manera:

Se describirán las ventajas de los modelos de calidad, seguidamente se identificarán puntos comunes y se listarán. La lista hallada se convertirá en los criterios para la elección de un Modelo de calidad para la Gestión de Requisitos.

#### **3.2.2.2 Elección del Modelo de calidad**

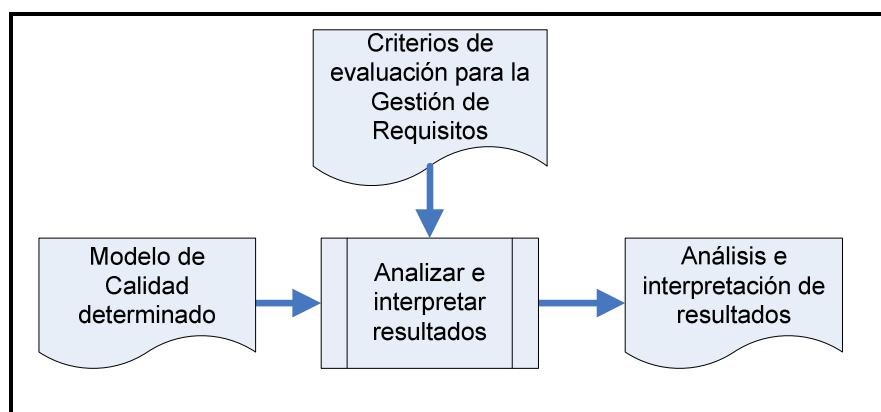
En la elección del modelo de calidad se seguirán los siguientes pasos:

- ❖ Definición del Problema
- ❖ Tabular los Modelos de calidad contra los factores que afectan la gestión de requisitos
- ❖ Incluir en la tabulación los criterios seleccionados y se procede con asignar un nivel de satisfacción a cada factor según criterio en base a la escala Likert
  - 1: Muy poco importante
  - 2: Poco importante
  - 3: Importancia media
  - 4: Algo Importante
  - 5: Muy importante
- ❖ Se calculará el resultado obtenido por cada Modelo al realizar la sumatoria de los puntajes obtenidos por cada factor en el paso anterior.
- ❖ El modelo mas adecuado para la realidad descrita será el Modelo que obtenga el mayor puntaje

### 3.3. Análisis e interpretación de los resultados

En esta última parte se realizará el análisis y la interpretación de los resultados en base a los criterios definidos para la Gestión de Requisitos

La figura 3-3 muestra como entrada el Modelo de calidad determinado y como control los criterios de evaluación para la Gestión de Requisitos para el análisis e interpretación de los resultados que dará como resultado el documento de análisis y la interpretación de los resultados.



**Figura 3-3 Factores que afectan la gestión de Requisitos**  
(Elaboración Propia)

### 3.4. Factores de Evaluación

Se toman los criterios de [30] en donde se detallan los factores de éxito de los SPI (Software Process Improvement).

Factores de éxito relacionados con la planificación se dividen en tres categorías según su contenido:

- Análisis de situación actual
- Definición de objetivos y
- Planificación de la mejora.

**Tabla 3-1 Factores de Éxito relacionados con la Planificación**

<b>Factores de Éxito</b>	<b>Sub-Factor de Éxito</b>	<b>Pregunta de Evaluación</b>
Análisis del estado actual	Estado actual del proceso es aclarado	¿Apoya el método a aclarar la situación actual del proceso?
Objetivo definición	Vínculo entre los objetivos de negocio y objetivos de mejoras establecido	¿Apoya el método de establecer un vínculo entre el negocio y las metas de mejoramiento?
	Objetivos de mejora son medibles	¿Apoyan el método a medir objetivos de mejora?
Mejora de la planificación	Un plan de mejora es generado	¿Apoya el método a la generación de un plan de mejora?

#### **4. CAPITULO IV: DESARROLLO PARA LA ELECCIÓN DEL MODELO DE CALIDAD PARA LA GESTIÓN DE REQUISITOS EN COLEGIOS PRIVADOS DE LIMA**

##### **4.1 Requisitos para desarrollo del caso de estudio**

Para la definición del método de selección, se necesita de:

- Tener los permisos necesarios para el acceso a información los audios de entrevistas al área de informática, que fueron realizadas mientras se recogía información para el mapeo de los procesos en la institución educativa particular de Lima
- Tener los permisos necesarios para el acceso a los diagramas de procesos del área de informática

##### **4.2 Problemática en la Institución educativa Bertolt Brecht**

En el colegio particular seleccionado como caso de estudio, es una institución relativamente nueva con solo 13 años de creada, cuenta actualmente con diferentes áreas administrativas y educativas y con una gran cantidad de alumnos en los diferentes niveles (Inicial, primaria y secundaria).

Al ser un colegio de gran tamaño cuenta con demandas de creación de nuevo software que solucione las necesidades administrativas y educativas, contando para esto con un área de informática y Estadística que presenta muchas carencias en el proceso de Gestión de Requisitos, debido a que no cuentan con un estándar o un Modelo de calidad de referencia a utilizar.

El proceso de desarrollo de nuevos aplicativos es el siguiente:

- Área solicitante presenta su lista de requerimientos
- El área de Estadística e Informática analiza los requerimientos
- El programador prepara el prototipo de desarrollo y el responsable de Estadística e informática desarrolla la propuesta

- En caso de que el área solicitante este de acuerdo se entrega toda la documentación al programador quien desarrolla la aplicación

Observando que la interacción entre el cliente (Área solicitante) y el responsable del análisis de los requisitos es mínima, además de la omisión de muchas actividades dentro del proceso que son establecidos por los Modelos de calidad, lo que produce modificaciones en el aplicativo, ambigüedades, productos software que al final no cumple con las expectativas para las cuales fue creado, etc.

#### **4.2.1 La institución educativa Bertolt Brecht**

- **Misión**

Brindar una formación integral a los estudiantes de los sectores populares, la cual les permitirá contribuir al desarrollo social del país.

- **Visión**

Ser una institución educativa referente en la comunidad por nuestra propuesta integral.

- **Objetivo**

Contribuir en la formación integral del educando fomentando desde los primeros grados la interiorización de sentimientos, conocimientos y orientaciones valorativas acordes con las reales necesidades sociales y en relación directa con la familia y la comunidad.



- **Reseña Histórica**

El colegio Bertolt Brecht inicia sus actividades en el año 1997, en un contexto de crisis social, con una propuesta alternativa de educación, la misma que se centra en la formación integral de la persona que le permita desempeñar un papel activo en el desarrollo de la sociedad.

El colegio Bertolt Brecht continúa la senda trazada por su promotor, el Instituto de Ciencias y Humanidades, de contribuir con el desarrollo social a través de una educación científica, democrática y nacional.

En su proceso histórico se identifican claramente dos momentos: el primero, de aplicación inicial de la propuesta educativa (1997-2003), y el segundo, de sistematización de la experiencia para la consolidación de la propuesta (2004 hasta la fecha).

### **4.3 Aplicación del método de selección del Modelo de Calidad**

#### **4.3.1 Primera etapa: Analizar el proceso de Gestión de Requisitos**

##### **4.3.1.1 Identificar necesidades en la Gestión de Requisitos**

Para la identificación de las necesidades en la Gestión de Requisitos se analiza todo el material recopilado y en base a la problemática presente se seleccionan las siguientes necesidades en la Gestión de Requisitos en el Colegio privado Bertolt Brecht.

La tabla 4-1 Necesidades en la Gestión de Requisitos, lista las necesidades.

**Tabla 4-1 Necesidades en la Gestión de Requisitos**

	Necesidades en la Gestión de Requisitos
N01	Programación de reuniones con cierta frecuencia para la elicitación de requisitos
N02	Control de cambios
N03	Validación formal de requisitos
N04	Procedimientos estándares
N05	Especificación formal de requisitos
N06	Actas de reunión y de aceptación
N07	Identificación de los usuarios principales para la Elicitación
N08	Estimación de recursos para el desarrollo del proyecto
N09	Elaboración de un cronograma y especificación de hitos de control
N10	Distribución de la carga laboral

#### **4.3.1.2 Identificación de factores según el método Scoring**

Para la identificación de los factores desarrollaremos el método Scoring, que consiste en:

##### **1. Identificación de la meta general del problema**

La meta es la siguiente: La identificación de los principales factores que apoyen a la elección del Modelo de Calidad más adecuado para la Gestión de Requisitos en Colegios privados.

##### **2. Identificación de alternativas**

Las alternativas son las necesidades en la Gestión de Requisitos definidos en el paso anterior.

**3. Listar los criterios a emplear en la toma de decisiones**

Los criterios a considerarse serán: La alineación con el negocio y puntos críticos en la Gestión de Requisitos.

**4. Asignar una ponderación para cada uno de los criterios**

Para este caso al contar con criterios de igual importancia para la Gestión de Requisitos se realiza una asignación de un peso = 1

**5. Establecer en cuanto satisface cada alternativa a nivel de cada uno de los criterios**

Para el establecimiento de los factores se evaluarán las necesidades obtenidas, así como lo muestra la tabla 4-2 Establecimiento de Factores, se le asignará una cantidad numérica se utilizará la escala Likert de 5 puntos para la evaluación de cada factor, en donde los puntajes se distribuyen de la siguiente manera:

- 1: Nada importante
- 2: Poco importante
- 3: Importancia media
- 4: Importante
- 5: Muy importante

Tabla 4-2 Establecimiento de Factores

Necesidades	Puntos Críticos en la Gestión de Requisitos	Criterio: Alineación con el Negocio	Puntaje
Programación de reuniones para la Elicitación de requisitos	5	5	10
Control de cambios	5	5	10
Validación formal de requisitos	5	4	9
Procedimientos estándares	4	4	8
Identificación de los usuarios principales para la Elicitación	3	4	7
Especificación formal de requisitos	3	3	6
Elaboración de un cronograma y especificación de hitos de control	3	3	6
Estimación de recursos para el desarrollo del proyecto	3	3	6
Distribución de la carga laboral	2	2	4
Actas de reunión y de aceptación	1	2	3

### 6. Calcular la puntuación de cada una de las alternativas

El cálculo de cada alternativa como lo muestra la figura 4-1 Resultado, el resultado es igual a la sumatoria de la multiplicación del peso del criterio (por ser criterios de igual importancia presentan como peso el valor de 1) por el valor de satisfacción asignado, de esta forma se cuantifica cada alternativa.

$$\text{Resultado: } \sum(\text{Peso})(\text{Valor de Satisfacción})$$

**Figura 4-1 Cálculo del Resultado**  
(Elaboración Propia)

### 7. Ordenar las alternativas en función de la puntuación

Se ordenan las puntuaciones de las necesidades de forma descendente y se toman las 5 primeras necesidades de las 10 listadas. Estas 5 primeras necesidades se convierten en los factores que apoyan en la elección del Modelo de Calidad más adecuado. La tabla 4-3 Tabla de factores, lista los factores seleccionados al aplicar el método Scoring.

**Tabla 4-3 Tabla de Factores**

	Factores
F01	Programación de reuniones para la Elicitación de requisitos
F02	Control de cambios
F03	Validación formal de requisitos
F04	Procedimientos estándares
F05	Identificación de los usuarios principales para la Elicitación

### 4.3.2 Segunda etapa: Elección de un Modelo de Calidad

#### 4.3.2.1 Selección de criterios

Para la selección de criterios se tabulan los Modelos de calidad y las ventajas de cada uno de ellos, seguidamente se revisan los puntos en común para obtener los criterios que se utilizarán para la selección del Modelo de Calidad. La tabla 4-4 Selección de Criterios, muestra las ventajas de cada uno de los Modelos de Calidad, y la obtención de los Criterios a partir de ellos.

<b>MODELOS</b>	<b>VENTAJAS</b>	<b>CRITERIOS</b>
<b>CMMI</b>	<ul style="list-style-type: none"> <li>• Mejora la comunicación</li> <li>• Mejora el ambiente de trabajo</li> <li>• Provee un enlace o relación mas explicita entre la Gestión y la Ingeniería con los objetivos del negocio</li> <li>• Guía paso a paso para la mejora, a través de niveles de madurez y capacidad</li> <li>• Transición del ‘aprendizaje individual’ al ‘aprendizaje de la organización’ por mejora continua, lecciones aprendidas y uso de bibliotecas y bases de datos de proyectos mejorados.</li> <li>• Dirigido a grandes proyectos</li> </ul>	<ul style="list-style-type: none"> <li>• Mejora la Comunicación</li> <li>• Sencillez del Modelo</li> <li>• Especificación de técnicas y procedimientos</li> </ul>
<b>METRICA3</b>	<ul style="list-style-type: none"> <li>• Adaptabilidad, Flexibilidad ,Sencillez</li> <li>• Definición detallada de técnicas y procedimientos</li> <li>• Facilitar la comunicación y entendimiento entre los distintos participantes en la producción de software a lo largo del ciclo de vida del proyecto</li> <li>• Dota a la Organización de productos software que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de requisitos</li> <li>• Dirigido para Pequeños y medianos proyectos</li> </ul>	
<b>ISO 12207</b>	<ul style="list-style-type: none"> <li>• Mejora la planificación</li> <li>• Mejor comunicación con los clientes y entre los profesionales de la entidad</li> <li>• Estándar Conocido y disponible</li> <li>• Modelo Concreto (Poco nivel de detalle)</li> <li>• Dirigido a grandes proyectos</li> </ul>	

**Tabla 4-4 Selección de Criterios**

La tabla 4-5 Criterios, muestra los criterios seleccionados para la evaluación del Modelo de Calidad

**Tabla 4-5 Criterios**

	<b>CRITERIOS</b>
C01	Mejora la Comunicación
C02	Sencillez del Modelo
C03	Especificación de técnicas y procedimientos

#### **4.3.2.2 Selección del Modelo de calidad**

Seleccionaremos el Modelo de Calidad con los siguientes pasos:

- ❖ Definición del Problema: La elección de un Modelo de calidad adecuado para el caso de estudio
- ❖ Seguidamente se tabulará los Modelos de calidad contra los factores que afectan la gestión de requisitos
- ❖ Incluir en la tabulación los criterios seleccionados y se procede con asignar un nivel de satisfacción a cada factor según criterio en base a la escala Likert 1: Nada importante; 2: Poco importante; 3: Importancia media; 4: Algo importante; 5: Muy importante
- ❖ Calcular la puntuación total obtenido por cada Modelo

En la tabla 4-6 Tabla Comparativa I, Observamos los niveles de satisfacción asignados y los resultados obtenidos por Modelos de Calidad.

**Tabla 4-6 Tabla Comparativa I**

<b>Factores</b>		<b>CMMI nivel 2</b>	<b>METRICA versión 3</b>	<b>ISO 12207</b>
Programación de reuniones para Elicitación de requisitos	C01	5	5	5
	C02	3	5	2
	C03	3	5	1
Control de cambios	C01	5	5	5
	C02	3	5	2
	C03	3	4	1
Validación formal de requisitos	C01	5	5	4
	C02	3	5	1
	C03	3	5	1
Identificación de los usuarios principales para la Elicitación	C01	5	5	5
	C02	3	4	3
	C03	3	4	1
Procedimientos estándares	C01	4	4	5
	C02	3	5	1
	C03	3	5	1
<b>Totales</b>		<b>54</b>	<b>71</b>	<b>38</b>

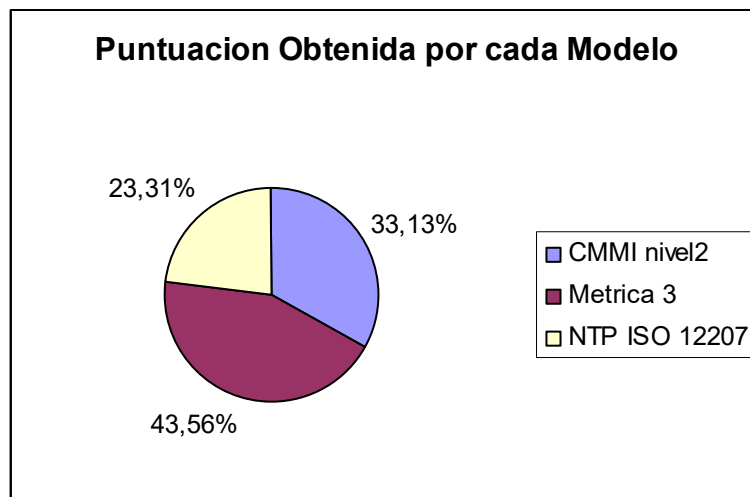


La tabla 4-7 tabla comparativa, muestra las sumas parciales que obtiene cada Modelo por cada factor. Esta suma parcial ayuda a la identificación de las fortalezas y debilidades de cada Modelo por factor.

**Tabla 4-7: Tabla comparativa**

<b>Factores</b>		<b>CMMI nivel 2</b>	<b>METRICA versión 3</b>	<b>ISO 12207</b>
Programación de reuniones para Elicitación de requisitos	C01	5	5	5
	C02	3	5	2
	C03	3	5	1
<b>Suma Parcial</b>		<b>11</b>	<b>15</b>	<b>8</b>
Control de cambios	C01	5	5	5
	C02	3	5	2
	C03	3	4	1
<b>Suma Parcial</b>		<b>11</b>	<b>14</b>	<b>8</b>
Validación formal de requisitos	C01	5	5	4
	C02	3	5	1
	C03	3	5	1
<b>Suma Parcial</b>		<b>11</b>	<b>15</b>	<b>6</b>
Identificación de los usuarios principales para la Elicitación	C01	5	5	5
	C02	3	4	3
	C03	3	4	1
<b>Suma Parcial</b>		<b>11</b>	<b>13</b>	<b>9</b>
Procedimientos estándares	C01	4	4	5
	C02	3	5	1
	C03	3	5	1
<b>Suma Parcial</b>		<b>10</b>	<b>14</b>	<b>7</b>

La figura 4-2, Puntuación obtenida por cada Modelo muestra gráficamente los porcentajes que obtienen los Modelos al ser evaluados.



**Figura 4-2 Puntuación obtenida por cada Modelo**  
(Elaboración Propia)

#### 4.4 Análisis e interpretación de los resultados

- ❖ El mayor puntaje es obtenido por el Modelo de Calidad: Métrica versión 3, que es el más adecuado para el caso de estudio, es el modelo que cumple con los criterios y factores seleccionados.
- ❖ De la tabla 4-7 Tabla comparativa, se obtiene que de los 3 Modelos de Calidad, Métrica versión 3 presenta ventajas sobre los demás Modelos en cada factor listado
- ❖ ISO 12207 al ser un Modelo Concreto, presenta mayores dificultades en la evaluación del criterio: Especificación de Técnicas y Procedimientos
- ❖ El Modelo CMMI nivel 2 presenta una puntuación Promedio en cada factor evaluado

- ❖ Las mayores fortalezas de Métrica Versión 3 son la Sencillez y la especificación de Técnicas y Procedimientos, que impactan directamente en los factores: Programación de reuniones para Elicitación de requisitos y Programación de reuniones para Elicitación de requisitos
- ❖ Métrica versión 3, al ser un Modelo Sencillo, con Técnicas y practicas detalladas y al estar dirigido a pequeños y medianos proyectos, se ajusta con la realidad descrita en el caso de estudio, haciéndola la elección más idónea

#### 4.5 Factores de Evaluación aplicados al Método de Selección

Se utilizan los factores de evaluación al proceso de la aplicación del Método de Selección, con el objetivo de asegurar que se el proceso de selección se encuentra dentro de los parámetros permitidos. La tabla 4-8 Aplicación de las preguntas, muestra los resultados de someter a evaluación el método de selección.

**Tabla 4-8 Aplicación de las preguntas**

<b>Pregunta de Evaluación</b>	<b>Respuesta</b>
¿Apoya el método a aclarar la situación actual del proceso?	Si, Al identificar las necesidades se aclara la situación actual y las deficiencias.
¿Apoya el método de establecer un vínculo entre el negocio y las acciones de mejoramiento?	Si, Al identificar las los factores, estos se toma considerando la alineación con el Negocio.
¿Apoyan el método a medir objetivos de mejora?	No, no ayuda en una medición
¿Apoya el método a la generación de un plan de mejora?	Si, la elección misma de un Modelo de Calidad implica la generación de un plan de mejora y de seguimiento.

La evaluación realizada nos indica que el Método de Selección aplicado cumple con más del 75% de los factores establecidos.

## 5. CONCLUSIONES

- ❖ Se eligió un Modelo de Calidad para la Gestión de Requisitos, en base a la comparativa entre Modelos de Calidad conocidos internacionalmente como: Métrica versión 3, CMMI nivel 2 y NTP ISO 12207
- ❖ Se Identificaron las necesidades en la Gestión de Requisitos de la institución educativa particular en base a la revisión del material recolectado que incluye: Audios de entrevistas, y diagramas de procesos.
- ❖ Se identificaron los factores principales que afectan la Gestión de requisitos para lo cual se utilizó el método del Scoring sobre las necesidades determinadas previamente.
- ❖ Se identificaron los criterios para la evaluación de los Modelos de Calidad en base a:
  - a. Primero: La identificación de las ventajas de cada Modelo de Calidad y
  - b. Segundo La identificación de los puntos en común en las ventajasEstos puntos en común se convirtieron en los criterios que fueron utilizados en la evaluación del Modelo.
- ❖ El Modelo de Calidad seleccionado es sencillo de entender y proporciona técnicas para la ejecución de cada actividad dentro del proceso de Gestión de Requisitos.
- ❖ El Modelo de calidad Métrica Versión 3, muestra un conjunto de técnicas y prácticas específicas de forma sencilla y fácil, orientándose a pequeños y medianos proyectos.
- ❖ El Modelo de calidad NTP ISO 12207 proporciona lineamientos más abstractos y no proporciona prácticas específicas, lo que lo convierte en un Modelo más complicado por la realidad descrita en el caso de estudio.

- ❖ Los Modelos de Calidad tomados como referencia para la presente tesina han sido seleccionados debido a que son los más conocidos dentro del ambiente de desarrollo de sistemas informáticos, y además porque la información acerca de esos Modelos de Calidad es abundante.
  
- ❖ Se sometió a evaluación el método de selección descrito obteniendo un 75% de satisfacción con respecto a los indicadores seleccionados.

## 6. RECOMENDACIONES

Una vez concluida esta tesina, se considera interesante investigar sobre otros aspectos relaciones al proceso de desarrollo del Software

- ❖ Extender los estudios expuestos en esta tesina, y abarcar otros aspectos que afectan el éxito de los proyectos software como: Análisis y Diseño de los sistemas software, las problemáticas y la elección de un Modelo adecuado para las instituciones educativas privadas
- ❖ Extender el campo de aplicación de la presente tesina y abarcar no solo instituciones educativas privadas, sino también nacionales.
- ❖ Complementar el Método de selección y transformarla en una herramienta software que permita la selección de Modelos de Calidad.

## BIBLIOGRAFIA

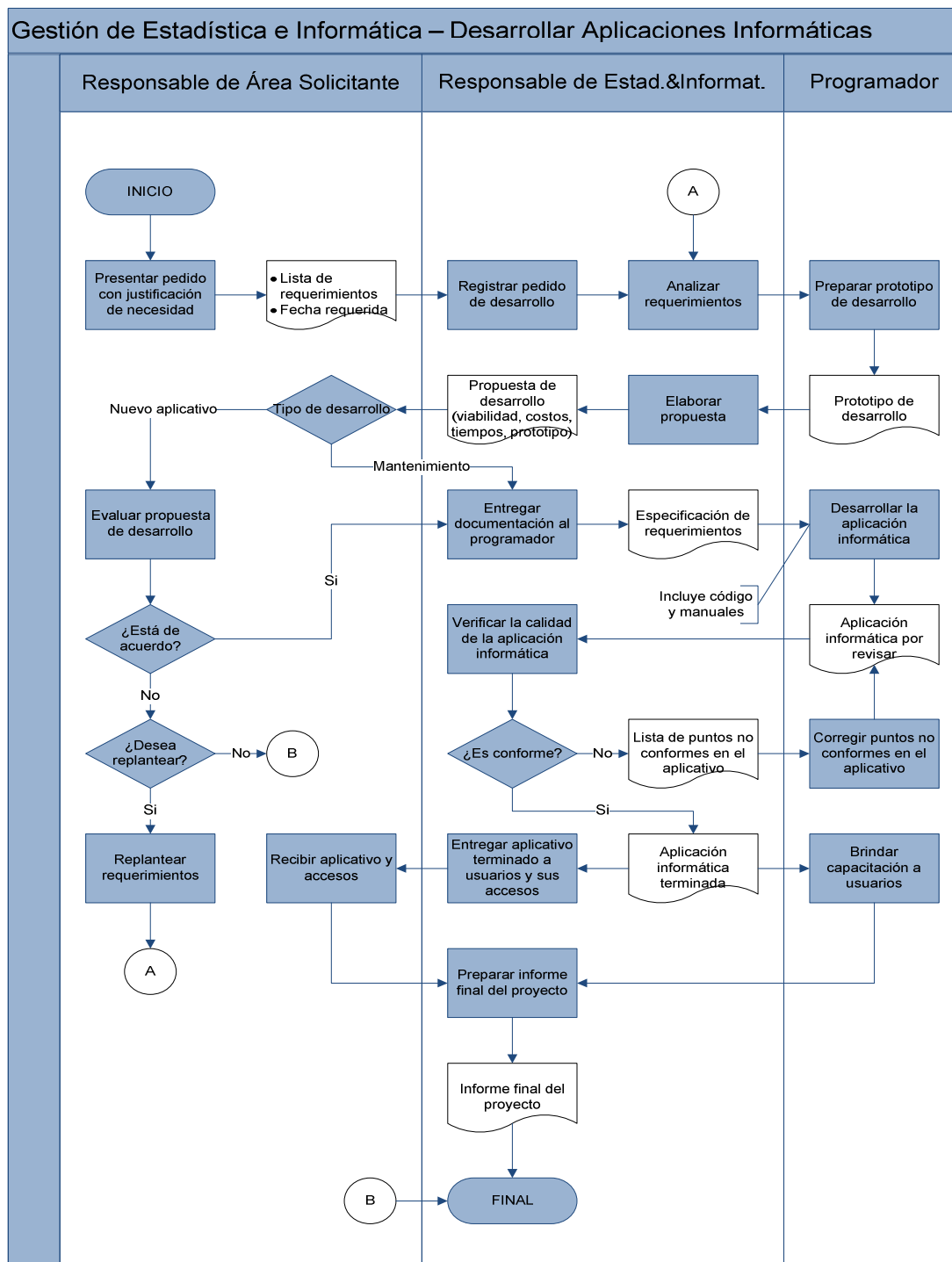
- [1] The Standish group. (1995). The Standish group Report Chaos, 1-8.
- [2] Grompone, J. (1996). Juan Grompone Gestión de Proyectos de Software.
- [3] IICD. (2007). Las TIC para el sector educativo, 94.
- [4] CUANTO. (2003). Situación de la Industria Nacional de Software en el Perú.
- [5] IEEE. (1990). IEEE Standard Glossary of Software Engineering. Terminology
- [6] Cháves, M. A. (2007). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software, *Vi*.
- [7] Sommerville, Ian. Ingeniería de Software (7ma Ed.) España: Pearson Educación, S.A, 2005. 687p.
- [8] Thomas, P. J. (2005). Definición de un Proceso de Elicitación de Objetivos, 106.
- [9] Roger S. Pressman. Ingeniería de Software un enfoque Práctico (5ta Ed.). España, Madrid, 2005. Concepción Fernández Madrid. 589p
- [10] Andriano, N. V. (2006). “Comparación del Proceso de Elicitación de Requerimientos en el desarrollo de Software a Medida y Empaquetado. Propuesta de métricas para la elicitación.”.
- [11] Gonzalez-baixauli, B., Laguna, M. A., & Sampaio, C. (1955). Aplicación de la Teoría de Constructos Personales a la Elicitación de Requisitos, 1-8.
- [12] Zapata, Carlos; Arango, F. (2004). Alineación entre metas organizacionales y Elicitación de Requisitos del Software.
- [13] Oliveros, A. (n.d.). *Elicitación de Objetivos a partir de Escenarios. Analysis*.
- [14] Villanueva, Isabel; Sanchez, Juan; Pastor, O. (2004). Elicitación de requisitos en sistemas de gestión orientados a procesos.
- [15] Aranda, G. N., Vizcaíno, A., Cechich, A., Piattini, M., & Soto, J. P. (n.d.). Una metodología para elicitación de requisitos en proyectos GSD.
- [16] Grande, J. I., & Martins, L. E. (n.d.). SIGERAR: Una Herramienta para Gerenciamiento de Requisitos.
- [17] Puello, P., & García, F. (n.d.). GERESOFT – Gestión de la Verificación y Validación de Requisitos. *Framework*.

- [18] Melendez, Nhur; Vega, Vianza; Gallardo, Jose; Meneses, C. (2009). IR-SIXSIGMA: Mejora de Calidad en Ingeniería de Requisitos Mediante la. Chile.
- [19] ISO/IEC. (2006). TECNOLOGIA DE LA INFORMACION. Procesos de ciclo de vida del Software.
- [20] Software Engineering Institute. (2002). Capability Maturity Model Integration (CMMI).
- [21] Gutiérrez, J. J., Escalona, M. J., Mejías, M., & Torres, J. (n.d.). *Mejora de la calidad de los requisitos mediante la generación de pruebas*.
- [22] Hadad, Graciela; Doorn, Jorge; Kaplan, G. (n.d.). Explicitar Requisitos del Software usando Escenarios.
- [23] Zapata, Carlos; Jaramillo, Aldrin; Arango, F. (2006). *Una propuesta para mejorar la completitud de requisitos utilizando un enfoque lingüístico*. *Analysis*, 0122-3461.
- [24] Ridao, M., & Doorn, J. H. (n.d.). *Estimación de Completitud en Modelos de Requisitos Basados en Lenguaje Natural*.
- [25] Estrada, H., Martínez, A., Pastor, O., & Sánchez, J. (n.d.). *Generación de Especificaciones de Requisitos de Software a partir de Modelos de Negocios : un enfoque basado en metas*. Mexico.
- [26] MÉTRICA. VERSIÓN 3. (n.d.). Análisis del Sistema de Información, 3.
- [27] Stelzer, D., & Mellis, W. (1999). Success Factors of Organizational Change in Software Process Improvement. *Change*, 4(4), 1-34.
- [28] Dybå, T. (2001). An Instrument for Measuring the Key Factors of Success in Software Process Improvement. *Industrial Research*, 357-390.
- [29] Halvorsen, C. P., & Conradi, R. (n.d.). A Taxiomatic Attempt at Comparing SPI Frameworks.
- [30] Komi, S. (2004). Development and Evaluation of Software Process Improvement. *Vtt Publications*, 535.
- [31] Rainer, A., & Hall, T. (1999). Key success factors for implementing software process improvement : a maturity-based analysis. *Journal of Systems and Software*, 1-20.
- [32] Ruiz, M. (n.d.). Modelos de Evaluación y Mejora de Procesos : Análisis Comparativo. *CICYT*, 1-18.



## ANEXO I. DIAGRAMA DE PROCESOS

Se presenta el diagrama del Procesos: “Desarrollar Aplicaciones informáticas”, de donde nos centraremos en la parte del proceso donde Gestionan los requisitos.



## ANEXO II. METODO DEL SCORING

### Método del SCORING

El método del Scoring es una manera rápida y sencilla para identificar la alternativa preferible en un problema de decisión multicriterio.

Las etapas del método son los siguientes:

1. Identificar la Meta General del Problema
2. Identificar las Alternativas
3. Listar los Criterios a emplear en la toma de decisión
4. Asignar una ponderación para cada uno de los Criterios
5. Establecer en cuanto satisface cada Alternativa a nivel de cada uno de los Criterios
6. Calcular el Score para cada una de las Alternativas
7. Ordenar las Alternativas en función del Score. La Alternativa con el Score más alto representa la Alternativa a recomendar.

Modelo para Calcular el Score:

$$S_j = \sum_i w_i r_{ij}$$

Donde:  $r_{ij}$  = rating de la Alternativa j en función del Criterio i

$w_i$  = ponderación para cada Criterio i

$S_j$  = Score para la Alternativa j