

**ASEGURAMIENTO DE LA CALIDAD EN EL PROCESO DELEGADO DE  
DESARROLLO DE SOFTWARE**

**EDWIN MARINO CASTILLO GIRALDO**

**UNIVERSIDAD EAFIT  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS  
MEDELLIN  
2006**

**ASEGURAMIENTO DE LA CALIDAD EN EL PROCESO DELEGADO DE  
DESARROLLO DE SOFTWARE**

**EDWIN MARINO CASTILLO GIRALDO**

**Trabajo presentado como requisito para optar al título de Ingeniero  
de Sistemas**

**Asesor  
NILSON YAIR MOLINA  
Ingeniero de Sistemas**

**UNIVERSIDAD EAFIT  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS  
MEDELLIN  
2006**

NOTA DE ACEPTACIÓN

---

---

---

---

Presidente del Jurado

---

Jurado

---

Jurado

## **AGRADECIMIENTOS**

El autor expresa su agradecimiento a:

A Dios, por darme la fortaleza y la sabiduría para realizar este trabajo.

A mis padres, cuyo apoyo constante e incondicional me ha permitido llegar al término de mi carrera.

A mi asesor, por el tiempo que me brindó, por apoyarme y guiarme en mi desarrollo profesional, por ser mi mentor, por ser mas que un asesor, por ser un amigo.

¡Infinitas Gracias!

## TABLA DE CONTENIDO

1.	DEFINICIÓN DEL PROBLEMA .....	1
2.	JUSTIFICACIÓN .....	2
3.	OBJETIVOS .....	4
3.1.	OBJETIVO GENERAL.....	4
3.2.	OBJETIVOS ESPECÍFICOS.....	4
4.	MARCO TEÓRICO.....	6
4.1.	La Tercerización.....	6
4.1.1.	Tipos de Outsourcing.....	7
4.1.2.	Clasificación por Estrategia de Aplicación.....	8
4.1.3.	Objetivos Estratégicos que Apoya el Outsourcing.....	9
4.1.4.	Razones para Subcontratar .....	11
4.1.5.	¿Cuándo se debe Subcontratar?.....	12
4.1.6.	Factores Críticos de Éxito cuando se Inician Procesos de Tercerización.....	14
4.1.7.	Desventajas del Outsourcing .....	15
4.1.8.	Outsourcing de Tecnologías de la Información.....	17
4.1.8.1.	Ciclo de vida del Outsourcing de TI.....	17
4.1.8.2.	Beneficios del Outsourcing de TI.....	21
4.1.8.3.	Riesgos en Outsourcing en TI.....	22
4.1.8.4.	Tipos de Outsourcing de TI.....	23
4.2.	LA CALIDAD .....	24
4.2.1.	Definición.....	24
4.2.2.	Calidad del Software.....	26
4.2.3.	Calidad Producto y Calidad Proceso .....	28
4.2.4.	Aseguramiento de la Calidad .....	29
4.2.5.	Factores de Calidad.....	30
4.2.5.1.	Eje 1: Operación del Producto.....	31
4.2.5.2.	Eje 2: Revisión de Producto .....	32
4.2.5.3.	Eje 3: Transición del Producto .....	33
4.3.	MODELOS Y ESTÁNDARES DE CALIDAD .....	35
4.3.1.	Norma ISO 9000.....	35
4.3.1.1.	Reseña Histórica .....	36
4.3.1.2.	La Familia ISO 9000.....	37
4.3.1.3.	Contenido de la ISO 9001 .....	40
4.3.1.4.	Los Ochos Principios .....	44
4.3.2.	Estándar ISO/IEC 15504.....	46
4.3.2.1.	La Necesidad de Otro Estándar .....	47
4.3.2.2.	Visión General de la Norma ISO/IEC 15504.....	48
4.3.2.3.	Centralización en la Valoración del Proceso.....	49
4.3.2.4.	Modelo Externo de Referencia de Proceso .....	50
4.3.2.5.	Marco de Referencia para la Medición .....	51
4.3.2.6.	Modelo para la Valoración de Procesos.....	55
4.3.2.7.	El Proceso de Valoración .....	56

4.3.3.	Modelos CMMI .....	59
4.3.3.1.	Disciplinas de CMMI .....	60
4.3.3.2.	Estructura de CMMI .....	61
4.3.3.3.	Componentes Requeridos, Esperados e Informativos.....	65
4.3.3.4.	Representaciones.....	66
4.3.3.5.	Comparativo entre ambas Representaciones .....	72
4.3.3.6.	Áreas de Proceso CMMI .....	73
4.4.	Gestión de Procesos.....	74
4.5.	Gestión de Proyectos .....	76
4.6.	Ingeniería .....	78
4.7.	Soporte .....	81
4.7.1.	Modelo CMMI-AM .....	84
4.7.2.	Estructura del Modelo.....	85
4.7.2.1.	Solicitud y Monitoreo de Contratos .....	86
4.7.2.2.	Transición a Operaciones y Soporte .....	88
4.7.2.3.	Prácticas Genéricas .....	89
4.8.	El Desarrollo de Software en el Contexto de la Delegación.....	90
4.8.1.	Relación entre el Cliente y el Proveedor.....	91
4.8.2.	Escenarios de un Proyecto de Desarrollo de Software .....	93
4.8.3.	Problemática de la Delegación .....	95
5.	MARCO DE PROCESOS PARA EL PROCESO DELEGADO DE DESARROLLO DE SOFTWARE .....	101
5.1.	Planeación y Organización del Proyecto .....	106
5.1.1.	Objetivos .....	107
5.1.2.	Actividades a Desarrollar .....	107
5.1.2.1.	Personalizar los Procesos de Adquisición .....	107
5.1.2.2.	Conformar Grupos para todos los Procesos de Adquisición.....	118
5.1.2.3.	Desarrollar el Plan de Adquisición .....	118
5.1.2.4.	Revisar el Plan de Adquisición y Aprobar .....	120
5.2.	Desarrollo de Requisitos .....	121
5.2.1.	La Asignación de Responsabilidades en el Desarrollo de Requisitos .	123
5.2.1.1.	Actividades del Proceso de Desarrollo de Requisitos.....	124
5.2.2.	Objetivos .....	126
5.2.3.	Actividades a Desarrollar .....	126
5.2.3.1.	Analizar el Problema.....	126
5.2.3.2.	Entender las Necesidades de Clientes y Usuarios .....	134
5.2.3.3.	Desarrollar Especificaciones Suplementarias.....	137
5.2.3.4.	Definir el Sistema .....	143
5.2.3.5.	Desarrollo de la Especificación de Requisitos.....	148
5.3.	Selección del Proveedor y Contratación.....	151
5.3.1.	Objetivos .....	152
5.3.2.	Actividades a Desarrollar .....	152
5.3.2.1.	Validar las Actividades de Solicitud.....	153
5.3.2.2.	Identificar Proveedores Potenciales .....	155
5.3.2.3.	Realizar Estimaciones de Tiempo y Costos .....	157
5.3.2.4.	Desarrollar y Enviar la Solicitud de Propuesta.....	158



## 1. DEFINICIÓN DEL PROBLEMA

En la actualidad se dispone de muchos marcos de trabajo y metodologías que ayudan a gestionar proyectos informáticos y a organizar el Proceso Software de las empresas desarrolladoras. Muchas de estas metodologías son aplicadas con éxito en las empresas de la industria del software ya que por naturaleza están orientadas a su objeto de negocio y son, relativamente, fácilmente adaptables a las necesidades y características de cada una. Dichas empresas obtienen grandes beneficios al identificar y optimizar sus procesos, tomando lo que requieren de las metodologías y marcos de trabajo existentes.

Por otro lado, están las empresas cuyo objeto de negocio no es el de desarrollar software, pero que se apoyan en las Tecnologías de la Información para soportar sus distintos procesos. Estas empresas, guiadas por motivaciones tales como la especialización en las tareas propias del negocio o la reducción de costos, tienden a tercerizar en menor o mayor grado muchos de sus procesos informáticos, entre ellos el desarrollo de software. La anterior situación se ha vuelto contraproducente para muchas empresas ya que han sido fuertemente afectados aspectos de los que antes se tenía pleno control: costos, plazos de entrega, calidad, soporte, garantías, servicio, fiabilidad, etc. En el caso del desarrollo de software, lo anteriormente descrito toma un matiz aún más negro ya que la inevitable atención por parte de las organizaciones a los problemas relacionados con los sistemas informáticos desarrollados por terceros hace que los objetivos de la tercerización se pierdan por completo y se incremente el costo de su mantenimiento.

Por este motivo, y por los distintos problemas que acarrea a las empresas del sector no software, este proyecto de grado pretende analizar con mayor detalle la problemática descrita, identificar y estudiar sus distintas perspectivas, y proponer unas guías de acción que ayuden a mitigarla.

## 2. JUSTIFICACIÓN

En la actualidad, donde los mercados son cada vez más globalizados y competitivos, las empresas tratan de ser más eficaces y eficientes en la gestión de sus procesos internos y externos, para así mantener su posición en dichos mercados. Una de las estrategias que a nivel gerencial ha ganado cada vez más importancia y que propende por la especialización en las tareas propias del negocio, es el de la tercerización de servicios. La tercerización es una tendencia que se viene aplicando con mucha fuerza desde la década de los 70's y que permite a las empresas concentrarse en los aspectos propios de su negocio. Esta práctica, que inicialmente se empezó a utilizar en actividades periféricas del negocio – limpieza, mantenimiento, transporte – ha pasado a aplicarse a procesos más importantes del mismo: Logística, Manufactura, Distribución, Marketing, y Tecnologías de la Información.

Pese a esta marcada tendencia del medio, la tercerización puede ser muy riesgosa ya que se puede llegar a situaciones poco deseables tales como la externalización de una actividad propia del negocio, la pérdida de control sobre actividades críticas, la generación de dependencia a proveedores, la mala comunicación y coordinación entre clientes y proveedores, y la renuencia del personal a este tipo de cambios. En el caso de la tercerización de servicios relacionados con las Tecnologías de la Información la situación es aún más crítica, ya que estos son cada vez más importantes para el negocio porque intervienen de forma directa en la cadena de valor y en los procesos de apoyo. Adicionalmente, existen servicios informáticos que se consideran confidenciales y que son difícilmente delegables a terceros.

En el caso concreto del desarrollo de software, la tercerización ha traído grandes inconvenientes a las organizaciones. Primero que todo, los sistemas de información desarrollados por terceros no siempre tienen la calidad que se espera y muchas veces se gasta más tiempo en ajustes y correcciones al

producto entregado que en el mismo desarrollo inicial. Como agravante, se tiene que el desgaste en el que se incurre por el re trabajo que debe realizarse en la detección y corrección de problemas tiene un impacto bastante negativo en los presupuestos asignados a las dependencias de Tecnologías de la Información. Por otro lado, los proveedores no siempre asumen su responsabilidad en los errores cometidos en los sistemas informáticos desarrollados y, por lo contrario, se aprovechan de esta situación para obtener más ganancias.

Son muchas las variables que entran en discusión cuando se trata de determinar los causales de esta situación. Primeramente, las áreas de Tecnologías de la Información (TI, para abreviar) no siempre cuentan con unas políticas bien definidas que regulen el Proceso Software<sup>1</sup> de la compañía y mucho menos con unos procedimientos que indiquen como se deben hacer las cosas. Muchas veces se desconocen o no se tienen establecidos los procesos al interior del Área de TI; no se tienen definidos roles ni responsabilidades; no se cuenta una metodología que sirva de guía, o simplemente la personas involucradas no tienen la habilidad requerida para desempeñar las labores propias del rol que desempeñan. En el peor de los casos, las consecuencias de la situación anteriormente descrita pueden llegar a afectar procesos claves del negocio, generando grandes pérdidas; en el mejor de los casos, estas consecuencias afectan procesos de apoyo al negocio. En todos los casos, la insatisfacción de los usuarios crece y las dependencias organizacionales de Tecnologías de la Información pierden credibilidad. En el literal “El Desarrollo de Software en el Contexto de la Delegación” se profundiza en este tema y se muestra evidencia estadística que da cuenta de la problemática.

---

<sup>1</sup> La literatura define al Proceso Software como “el conjunto de personas, estructuras organizacionales, reglas, políticas, actividades o procedimientos, componentes de software, metodologías y herramientas utilizadas o creadas específicamente para conceptualizar, desarrollar, ofrecer servicios, innovar o extender un producto de software...” [GOR95].

### 3. OBJETIVOS

#### 3.1. OBJETIVO GENERAL

Establecer una estrategia ágil que permita gestionar eficientemente el desarrollo delegado de software, que sirva como referencia a las empresas y que permita entregar a los usuarios productos informáticos de alta calidad y que satisfagan de forma concreta y confiable las necesidades de negocio.

#### 3.2. OBJETIVOS ESPECÍFICOS

- Identificar y documentar el contexto de la tercerización de TI, sus ventajas y desventajas, su utilización como estrategia corporativa, las debilidades y oportunidades ofrecidas, y establecer los aspectos de esta estrategia que impactan la problemática descrita.
- Desarrollar un marco teórico en el cual se base la estrategia que se propondrá para hacer frente a la problemática descrita. Este marco teórico debe incluir los estándares, las metodologías, los marcos de trabajo y demás elementos que se utilicen en la propuesta desarrollada.
- Identificar los procesos o áreas de proceso<sup>2</sup> que desde el punto de vista del desarrollo delegado de software deben trabajarse con mayor prioridad para mitigar la problemática descrita.
- Establecer una propuesta metodológica basada en procesos y procedimientos que trabaje las áreas de proceso que más tienen impacto en el proceso delegado de desarrollo de software y que apoye los objetivos de la tercerización.

---

<sup>2</sup> Según la definición del SEI un área de proceso es un conjunto de prácticas relacionadas en un área que, cuando se trabajan colectivamente, satisfacen un conjunto de objetivos considerados como importantes para lograr un mejoramiento significativo en esa área. [SEI02].

- Establecer una propuesta metodológica basada en procesos y procedimientos que haga control efectivo de los procesos establecidos y que identifique oportunamente los problemas que se puedan presentar.

## 4. MARCO TEÓRICO

### 4.1. La Tercerización

El outsourcing, también llamado tercerización, subcontratación o externalización, puede definirse como la delegación de funciones u operaciones organizacionales “no vitales” a una entidad externa que se especializa en dichas funciones [WIK06]. El término “no vitales” tiene una connotación especial, y hace referencia a las operaciones o funciones de la organización que hacen parte de la cadena de valor o por medio de las cuales estas obtienen un beneficio directo. Por lo general, una organización no terceriza este tipo de funciones ya que de ellas depende en gran parte su subsistencia.

Son muchas las definiciones que se han generado en torno al tema de la tercerización y muchos son los nuevos conceptos que han surgido con base en el. A continuación se describen algunas de estas definiciones:

“... La transferencia a una tercera parte de la responsabilidad gerencial permanente para la provisión de un servicio, regida por un acuerdo de nivel de servicio...” [CAC04].

“...La transferencia de una función o funciones comerciales internas, más cualquier activo asociado, a un proveedor externo o proveedores de servicios que ofrece un servicio definido durante un periodo específico de tiempo, a un precio acordado, si bien probablemente limitado...”<sup>3</sup>.

“...La acción de recurrir a una agencia exterior para operar una función que anteriormente se realizaba dentro de una compañía...”<sup>4</sup>.

---

<sup>3</sup> [HEY02], p. 27.

<sup>4</sup> [ROR96], p. 4.

“... Adquisición sistemática, total o parcial y mediante proveedores externos, de ciertos bienes y servicios necesarios para el funcionamiento operativo de una empresa, siempre que hayan sido producidos por la propia empresa o esta se halle en condiciones de hacerlo y se trate de bienes o servicios vinculados a su actividad...”<sup>5</sup>.

A pesar de la multiplicidad de definiciones, debe tenerse siempre presente que la esencia es la misma: delegar en proveedores funciones u operaciones organizacionales que no hacen parte del núcleo del negocio. Según [BAZ06], el outsourcing puede ser visto como el proceso de transferir una función de negocio existente a un proveedor con el fin de utilizar estratégicamente los recursos externos para desarrollar las actividades que anteriormente se desarrollaban en casa. Es importante aclarar, que cuando se habla de transferir una función de negocio, se está hablando también de transferir los recursos humanos y físicos requeridos para desempeñar esa función, es decir, la gente y los materiales. El outsourcing involucra, además, la transferencia de una parte significativa de la gestión administrativa y de la toma de decisiones que se realizaban sobre la operación delegada. Debe tenerse en cuenta, que la compra de productos provenientes de una organización externa no es outsourcing; esto simplemente es una relación comercial con un vendedor. De manera similar, se tiende a confundir mucho la adquisición de servicios proveídos por entes externos con el outsourcing, cuando este solo puede considerarse como tal, cuando existe una considerable carga entre las partes en términos del intercambio de información, coordinación y confianza.

#### **4.1.1. Tipos de Outsourcing**

El outsourcing puede ser tipificado de muchas maneras dependiendo de variables tales como el tamaño de la organización, su capacidad financiera y

---

<sup>5</sup> [CAC04], p. 9.

sus necesidades. Desde el punto de vista de los acuerdos que una organización puede establecer con actores externos, se pueden establecer las siguientes tipologías [CAE05]:

*Outsourcing total:* este tipo de outsourcing implica la transferencia de la organización contratante al ente contratado tanto de los activos humanos como de los físicos. Lo anterior puede referirse a equipos, personal, redes, operaciones y responsabilidades administrativas.

*Outsourcing parcial:* es cuando solo se transfieren algunos de los elementos mencionados anteriormente.

*Outsourcing en áreas tecnológicas o funcionales:* hace referencia a la externalización de funciones relativas a la gestión de las tecnologías duras (hardware) y de las tecnologías blandas (software).

*Por lugar donde se desarrolla:* hace referencia a la ubicación geográfica donde se desarrolla la prestación de los servicios contratados.

*Por la duración:* hace referencia a la duración o extensión en tiempo del contrato de prestación de servicios.

*Por los costos:* esta tipificación evalúa con precisión la relación Costo / Beneficio para determinar si realmente es más favorable tercerizar.

*Por pertenencia de activos:* se describe como será la participación de las partes en el suministro de los activos requeridos para habilitar la prestación de los servicios.

#### **4.1.2. Clasificación por Estrategia de Aplicación**

El proceso de tercerización, es decir, el proceso mediante el cual se busca subcontratar la ejecución de algunas operaciones o funciones organizacionales, puede llevarse a cabo mediante la aplicación de distintas estrategias. La escogencia de una estrategia de implementación de un proceso de tercerización, depende exclusivamente de las circunstancias particulares de

cada organización. A continuación se describen brevemente algunas de esas estrategias [FGR98]:

*Outsourcing Integral:* hace referencia a la contratación de un tercero para que asuma todas las operaciones o funciones que una dependencia o unidad organizacional desempeña.

*Outsourcing Selectivo:* hace referencia a la tercerización de un conjunto limitado de funciones u operaciones organizacionales, considerados como no neurálgicos.

*Outsourcing Progresivo:* es pasar de un conjunto limitado de procesos delegados a un tercero a la delegación total de los mismos, es decir, es la evolución paulatina del outsourcing selectivo al outsourcing integral.

*Smartsourcing:* en este tipo de estrategia el tercero suministra a la organización contratante el recurso humano requerido para llevar a cabo la operación o función delegada. Debe tenerse en cuenta que el recurso asignado se supedita a las condiciones y requerimientos de la organización contratante, y el tercero se encarga de realizar los pagos de nómina, la motivación y el entrenamiento del personal.

#### **4.1.3. Objetivos Estratégicos que Apoya el Outsourcing**

Los objetivos estratégicos son aquellos objetivos que se plantean para el largo plazo y que buscan definir la situación futura de la organización [LOP06]. Este tipo de objetivos establecen la ruta a seguir por una compañía en los años venideros, y son una de las decisiones más críticas e importantes que los directivos deben tomar. Bien planteada, una estrategia de tercerización puede entrar a apoyar dichos objetivos estratégicos, dado que los beneficios que se

pueden obtener de esta, repercuten en variables sumamente relevantes para cualquier organización. Algunos de estos objetivos pueden ser<sup>6</sup>:

- Lograr la efectividad enfocándose en lo que la organización mejor sabe hacer
- Promover una organización flexible capaz de adaptarse a las circunstancias del mercado, las variables del entorno y la situación tecnológica
- Incrementar el desempeño organizacional a través de una mayor calidad de los productos y/o servicios ofrecidos, el aumento de la productividad, ciclos de producción más cortos, entregas puntuales, optimización de los recursos, y disponibilidad
- Transferir el costo del personal (incluyendo los beneficios de ley) y los gastos administrativos al contratista
- Convertir los costos fijos en costos variables
- Obtener ahorros en gastos relativos a equipos, personal, inventarios y similares, para liberar esos recursos para la organización
- Generar oportunidades de negocio y permear el mercado a través del establecimiento de una red de proveedores
- Disponer de los recursos financieros y la flexibilidad necesarias para expandir las operaciones del negocio
- Obtener conocimientos y recibir sugerencias que de manera innovadora pueden ayudar a mejorar los productos y/o servicios ofrecidos por la organización
- Mejorar la imagen, confianza y credibilidad organizacionales como consecuencia indirecta de la asociación con proveedores reconocidos

---

<sup>6</sup> [CAE05], p. 56.

#### 4.1.4. Razones para Subcontratar

La figura 4.1 sintetiza las razones por las cuales las organizaciones deciden subcontratar. Básicamente, existen dos grandes motivadores. El primero de ellos y, como era de esperar, tiene que ver con la reducción de costos. Los costos operacionales son uno de los factores que más importancia ha cobrado para las organizaciones en la actualidad y, a partir de ellos, miden su desempeño y el beneficio obtenido. En segundo lugar, esta la especialización. La necesidad de ser cada vez más competitivos a obligado a las organizaciones a ser mas productivas; la focalización en las actividades neurálgicas del negocio es requisito indispensable para alcanzar la eficiencia operacional. Algunas ventajas de delegar en terceros actividades no críticas son:

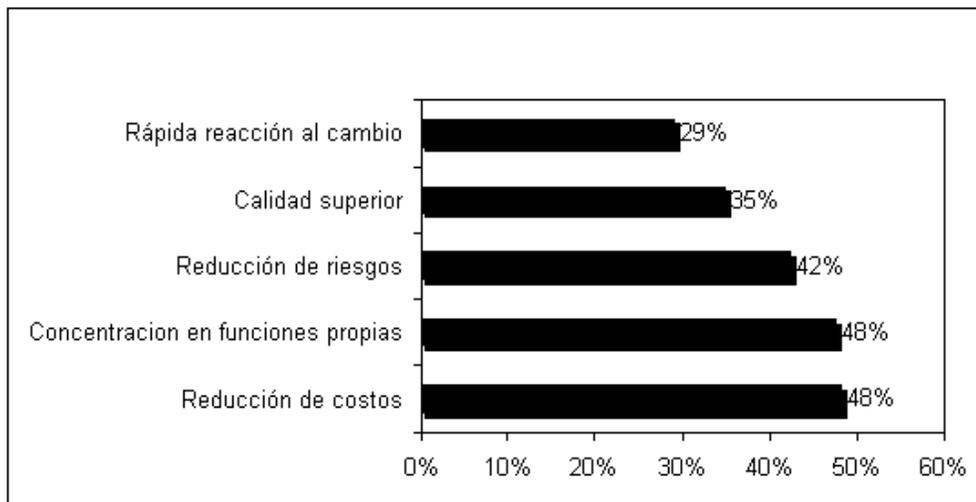


Figura 4.1. Razones para adoptar el outsourcing. Fuente [MOG06].

- Reducir muchos de los gastos en los que se incurre en las labores operativas de la organización
- En algunas organizaciones ayudar a mantener una mejor gestión empresarial
- Aumentar la ventaja competitiva y la diferenciación frente a las compañías de la competencia; esto lleva a las compañías a responder rápidamente a los cambios del entorno

- La tercerización permite tener mas control sobre las tareas u operaciones que anteriormente estaban descuidadas
- Obtener mayor eficiencia, puesto que la organización se concentra mas en las labores que tienen que ver con su objeto de negocio
- Se pueden destinar recursos para otros fines
- Se tiene la posibilidad de contar con un grupo de especialistas en el tema, los cuales pueden enfrentar con mayor facilidad los inconvenientes y plantear soluciones mas acertadas
- Habilita a las organizaciones para ser líderes en su negocio, ya que se vuelven mas eficientes al olvidarse definitivamente de tareas no críticas que les demandaban mucha atención
- Ayuda a disminuir los gastos en infraestructura o por lo menos ayuda a compensarlos con lo que se paga a la empresa contratista
- Lleva a la empresa a redefinir sus procesos, esto implica cambios en algunas de sus políticas
- La empresa puede disponer de sus empleados para que realicen otras labores que implican más cuidado y dedicación para la compañía
- Permite llevar una mejor planificación de las actividades, incrementando la capacidad de la organización para entender el funcionamiento del negocio y la capacidad para alcanzar los objetivos
- Las organizaciones pueden sacar provecho de adelantos tecnológicos sin necesidad de adquirirlos o de entrenar al personal

#### **4.1.5. ¿Cuándo se debe Subcontratar?**

Son muchos las razones válidas que se podrían tomar en consideración para tomar la decisión de delegar funciones u operaciones organizacionales. Como se mencionó anteriormente, todo aquello que tenga que ver con reducción de costos operacionales y especialización en las funciones propias del negocio, se convierte en un argumento de peso para promover una estrategia de outsourcing. Adicionalmente, toda organización se ve afectada por una serie de

factores o circunstancias internas y/o externas que pueden convertirse en criterios válidos para delegar una función u operación de negocio <sup>7</sup>:

- Cuando existen grandes desfases de tiempo en los proyectos
- Cuando la función u operación a ejecutar requiere de gran esfuerzo o tiempo
- Cuando la tecnología se encuentra rezagada y no puede responder de manera oportuna a las necesidades de negocio ni a la dinámica organizacional
- Cuando se presentan sobrecostos en el mantenimiento de sistemas legados<sup>8</sup> respecto a los costos típicos de mantenimiento
- Cuando se requiere estandarizar o integrar muchos sistemas
- Cuando se presentan cambios estructurales en la tecnología
- Cuando de manera repetitiva y constante la calidad de los productos o servicios propios no es satisfactoria
- En ausencia de un plan que indique como alcanzar los objetivos estratégicos de la organización
- Cuando la experticia interna no es suficiente para adelantar funciones u operaciones organizacionales
- Para crear un centro de consolidación de datos
- Cuando los costos fijos se incrementan y las utilidades desmejoran
- Cuando la organización se encuentra en alguna transición o se presentan eventos coyunturales
- Cuando la directriz es reducir los costos y aumentar las utilidades
- Cuando los sistemas de información organizacionales generan unos costos fijos muy elevados
- Cuando se están desempeñando procesos completamente ajenos al objeto de negocio de la organización

---

<sup>7</sup> [CAE05], p. 50.

<sup>8</sup> Según [GRO03], el término sistema legado se refiere a sistemas software de gran envergadura, monolíticos y complejos por naturaleza, los cuales han sido escritos en lenguajes de programación procedimentales, tales como COBOL, PASCAL, FORTRAN, C o similares.

#### **4.1.6. Factores Críticos de Éxito cuando se Inician Procesos de Tercerización**

Son muchas las variables que deben tenerse en cuenta a la hora de iniciar un proceso de tercerización. Aspectos tales como la capacidad del proveedor, su reconocimiento e historia en el mercado, su infraestructura técnica, sus estrategias de contingencia, su plan de soporte, y similares, se convierten en factores críticos que pueden determinar el éxito o fracaso de una estrategia de tercerización. Este tema es tratado con mayor detalle en el literal “Selección del Proveedor y Contratación” (ver sección 5.3) que hace parte de la estrategia propuesta en este trabajo. A continuación se describen brevemente algunos de los aspectos que podrían entrar en consideración:

- Ubicación geográfica de la compañía que presta el servicio
- Compromiso con la privacidad
- Disponibilidad del contratista para prestar el servicio
- Contar con la tecnología y los equipos requeridos para lograr un óptimo desempeño
- Portabilidad de los productos o servicios ofrecidos
- Niveles de servicio ofrecidos
- Estrategia para el manejo de las comunicaciones y el reporte de incidentes y de los servicios ofrecidos en concordancia con los acuerdos establecidos
- Contar con planes de contingencia para tomar decisiones ante situaciones inesperadas
- Contar con planes de riesgos y planes de recuperación
- Estrategias para el entrenamiento de clientes y usuarios
- Calidad del producto o servicio ofrecido
- Proyectos anteriores en los que han participado con otras compañías (experiencia) y que han sido exitosos
- Madurez y trayectoria de la empresa contratista en el campo objeto de la contratación

#### 4.1.7. Desventajas del Outsourcing

Como era de esperar, todo lo bueno tiene sus cosas malas, y en el caso del outsourcing lo anterior es aún mas cierto. La figura 4.2 resume y porcentualiza las principales desventajas del outsourcing. Las desventajas y problemas adicionales que podrían presentarse en el desarrollo de una estrategia de tercerización o en la ejecución de la misma son descritos en los ítems inferiores [MOL06]:

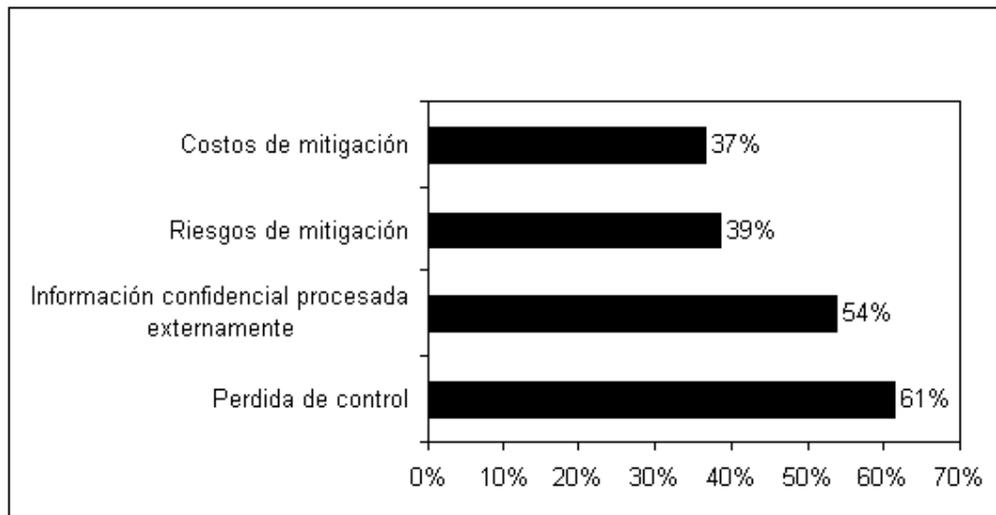


Figura 4.2. Desventajas del Outsourcing. Fuente [MOG06].

- Puede generarse una alta dependencia con la empresa contratista, hasta el punto que la organización puede perder su autonomía para desempeñar las funciones internas
- Pérdida de control
- Pérdida de la privacidad debido a que la organización debe proporcionar información confidencial al contratista
- La falta de experiencia del proveedor puede poner en riesgo el logro de los objetivos del proyecto
- Altos costos, debido a que solo se pueden contratar los servicios de organizaciones reconocidas y altamente especializadas
- El talento experto que hay en la organización se empieza a diseminar: pérdida del “know-how”

- La estabilidad financiera del proveedor puede cambiar y los compromisos suscritos con la organización contratante pueden verse afectados
- Dado que por lo general los proyectos de tecnología se caracterizan por ser riesgosos y por un alto grado de incertidumbre, pueden presentarse cambios en el negocio o en la tecnología que no fueron inicialmente contemplados en el contrato
- Incompatibilidad en formas de pensar, en ideologías del cliente y del proveedor
- Se pueden presentar cambios en el entorno que no pueden ser controlados con anticipación
- Existe el riesgo que el contratista pueda convertirse en proveedor de la competencia al hacer un uso poco ético de la información y/o conocimientos adquiridos en los trabajos desempeñados con la organización contratante [CAE05]
- Cuando las funciones u operaciones delegadas en el proveedor pueden volver a brindar una ventaja competitiva para la organización, es difícil reasumir el control de dichas actividades dado el alto costo asociado a esa decisión
- Si se requiriera cambiar el proveedor inicialmente seleccionado porque no llena las expectativas planteadas, el costo de esta decisión podría ser muy elevado
- Los ahorros esperados por concepto de outsourcing puede que no sean los esperados
- La organización podría perder contacto con nuevas tecnologías que le brindarían oportunidades para mejorar o innovar los productos y servicios ofrecidos por la misma

#### 4.1.8. Outsourcing de Tecnologías de la Información

"La prestación y administración de los servicios de TI se puede subcontratar, pero la responsabilidad y supervisión de los mismos debe permanecer en la empresa que externaliza esos servicios" [HIS05]. La anterior frase refleja el pensamiento de organizaciones que, tras años de aprendizaje y de haber enfrentado un sin número de problemas y experiencias en cuanto a la delegación de servicios de Tecnologías de la Información (TI, para abreviar) , han logrado alcanzar un nivel de madurez adecuado que les permite gestionar estos temas con eficacia y eficiencia.

El Outsourcing de TI (ITO, por sus siglas en inglés) se puede definir como el proceso de delegar parcialmente o en su totalidad las funciones de TI a uno o más proveedores [EUR04].

##### 4.1.8.1. Ciclo de vida del Outsourcing de TI

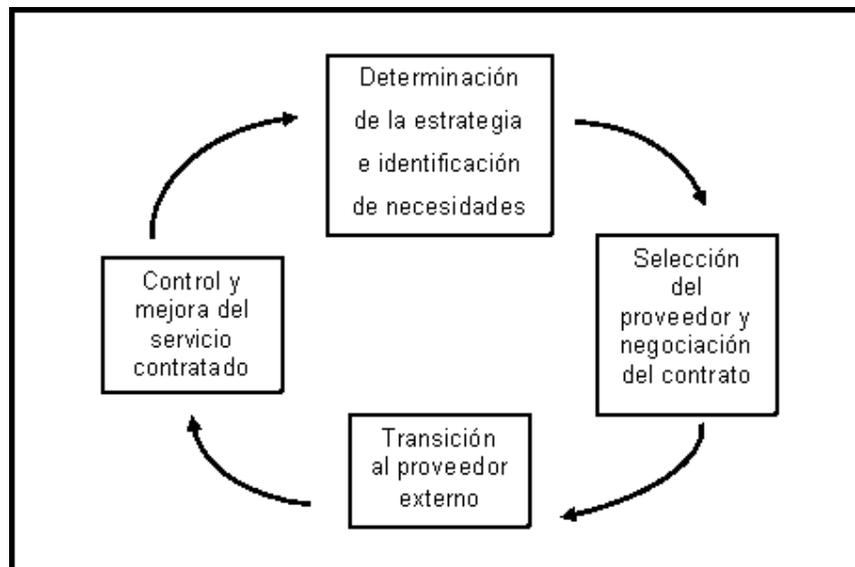


Figura 4.3. Ciclo de vida del outsourcing de TI. Fuente [MOG06].

La figura 4.3 resume las fases que componen el ciclo de vida del outsourcing de tecnologías de la información. Estas fases son [MOG06]:

- Determinación de la estrategia e identificación de las necesidades
- Selección del proveedor y negociación del contrato
- Transición al proveedor externo
- Control y mejora del servicio contratado

En la primera fase se identifican y especifican las necesidades y requerimientos por los cuales se escogió establecer una estrategia de TI; también se analiza si dicha estrategia es realmente adecuada. En la segunda fase se lleva a cabo todo el proceso de búsqueda, evaluación, selección y contratación del proveedor. La fase tres tiene como objetivo realizar la transición de la organización contratante al proveedor de la(s) función(es) u operación(es) delegada(s), lo cual incluye los recursos humanos y físicos que hayan sido pactados. La última fase tiene que ver con los mecanismos que la organización contratante aplica para verificar y asegurar que el proveedor si esta cumpliendo con los acuerdos suscritos y, en caso de lo contrario, aplicar las acciones correctivas que sean necesarias.

El outsourcing de TI según el Gartner Group, es una relación entre un cliente y un proveedor para contratar una actividad a desarrollar que se requiere para apoyar las funciones de tecnología de información de una empresa, normalmente se incluye:

- Gestión de aplicaciones
- Gestión de centros de datos
- Operaciones con redes LAN, WAN
- Soporte técnico a los usuarios de los sistemas de la compañía
- Servicio de Help Desk
- Soporte de la gestión del servicio contratado

La subcontratación de este tipo de servicios es una labor muy compleja que requiere de una buena planeación de todas las actividades, además de la documentación de los procesos de negocio adscritos a TI. El outsourcing de TI ocupa el mayor porcentaje en prestación de servicios dentro de las

organizaciones<sup>9</sup>, lo cual refleja la importancia de este tema. Los proveedores de servicios deben estar muy preparados porque la competencia es fuerte, deben responder a sus clientes con servicios de calidad y al mismo tiempo con costos asequibles, deben contar con una buena gestión y administración del negocio que les permita garantizar sus servicios.

Muchos de los proyectos de TI no se llevan a cabo o fallan en alguna de sus etapas, debido en gran parte a la mala planificación de los recursos tanto económicos como de tiempo, se realizan análisis de presupuestos muy apretados, no se hace un análisis adecuado de los riesgos del proyecto y por tanto no se incluyen las holguras requeridas dentro de los cronogramas, y no se tiene en cuenta la complejidad que implica trabajar con un proyecto de TI donde muchas de las variables son inciertas o dependen de factores externos que no es posible controlar.

Gartner “lanzó un reporte que delineó las fallas comunes de las compañías que subcontratan servicios de TI. El reporte identificó cinco factores que a menudo fallan en las compañías para considerar por completo cuando deciden si subcontratan o no las funciones de TI: costo, productividad, comunicaciones, cultura y buena disposición organizacional” [LEM06].

*Los costos* son una variable que es cambiante durante la trayectoria de un proyecto y es por esto que debe analizarse muy bien antes de iniciar con el mismo. Es importante que tanto los directivos como el personal de la dependencia organizacional de tecnología o sistemas de información, realicen un estudio de todas las variables que pueden influir sobre el costo, al igual que se debe evaluar todos los escenarios que pueden o no presentarse durante el proyecto, para de esta forma saber como enfrentarlos. Hay algo claro en todos los proyectos, y es que, al inicio los costos siempre serán más altos que al finalizar el proyecto. Esto, siempre y cuando, exista una buena planeación del

---

<sup>9</sup> [PER04], p. 2.

mismo y que el retorno de la inversión (ROI, por sus siglas en inglés) puede no manifestarse en poco tiempo.

*La productividad* es un aspecto que también puede variar o cambiar a lo largo del desarrollo del proyecto; en realidad, es una variable ambivalente porque pueden presentarse circunstancias en las que los niveles de productividad pueden elevarse y otras en las cuales estos pueden decaer considerablemente. Esto en gran parte porque el contratar un proveedor implica grandes cambios organizacionales; por ejemplo, puede implicar la rotación de puestos para algunos empleados lo cual puede afectar la productividad o el tiempo requerido por los proveedores para familiarizarse con el negocio, pone en riesgo la eficiencia de los mismos.

*La comunicación* resulta un factor determinante en la subcontratación porque es muy importante que las personas que están involucradas con TI y también los directivos que se encuentren comprometidos, entiendan claramente la función que van a desempeñar con el outsourcing. Es indispensable que todos participen activamente en el proceso, para que de esta forma no se vayan a presentar inconformidades una vez el proyecto este marchando. Se debe adoptar una metodología donde se documenten todos los procesos y las modificaciones que se vayan a llevar a cabo, esto para evitar las ambigüedades y los malos entendidos.

*La cultura* es un factor que se debe considerar para mantener una relación laboral apropiada entre las partes involucradas en el proyecto. La cultura, según Gartner, difiere entre un país y otro y por eso es importante tener en cuenta este aspecto a la hora de contratar al proveedor. La cultura organizacional incide sobre muchas de las decisiones y por eso, antes de seleccionar un proveedor, debe indagarse acerca de sus metodologías, sus estándares y políticas, y similares.

Los proveedores se encargan de decisiones de tipo tecnológico, gestión y administración del proyecto, operaciones con las infraestructuras informáticas y similares. Puede ocurrir en algunas ocasiones que la experiencia que tienen muchos de los proveedores con las compañías que los contratan algunas veces no son tan gratificantes, porque pueden sobrepasar el presupuesto que había sido asignado, muchas inversiones y pocos resultados beneficiosos, pueden no tener disponibilidad de todos los recursos y lo que es peor la insatisfacción por parte de los miembros de la compañía ante las metodologías aplicadas.

#### **4.1.8.2. Beneficios del Outsourcing de TI**

Estos son algunos de los beneficios del outsourcing de TI:

- Gestionar eficientemente los recursos
- Aprovechar las habilidades y el conocimiento de los empleados en las áreas en las cuales son fuertes
- Integrar la cadena de suministros a las nuevas soluciones propuestas
- Integrar las capacidades de gestión financiera para proporcionar un único conjunto de datos para todos los reportes
- Ayuda a mejorar la productividad
- El outsourcing de TI puede traer beneficios para el negocio como tal y puede llevar a la empresa a mantenerse actualizada en cuanto a la tecnología puesto que se cuenta con el conocimiento de expertos que los pueden asesorar
- Para las compañías en la cuales no se tiene muy definida una metodología de trabajo, el outsourcing puede convertirse en apoyo para mejorar la forma de trabajo y por lo tanto el desempeño de las personas involucradas al interior de la compañía
- Disminución de la inversión en nuevas tecnologías porque la empresa contratista debe proveer toda esta infraestructura

- Calidad del servicio garantizada
- Mayor flexibilidad cuando se requiere cambiar requerimientos
- Aprender nuevas ideas de los proveedores para empezar a incorporarlas en la compañía
- Actualización tecnológica permanente
- Se reducen los gastos en que se incurre al tener que capacitar al personal de la compañía contratante
- Mejor procesamiento de los datos y manejo de la información

#### **4.1.8.3. Riesgos en Outsourcing en TI**

Estos son algunos de los riesgos mas importantes que deben considerarse al establecer una estrategia de outsourcing de TI:

- Accesos no autorizados a la información que es confidencial para la compañía
- Llevar a cabo una mala gestión e inadecuada administración por parte de la empresa contratista
- Mala adecuación de los sistemas y de las redes para realizar las operaciones
- Falta de compromiso por parte de los directivos de la compañía ante el negocio y poco apoyo en la toma de decisiones
- Resistencia al cambio por parte de empleados de la organización
- Falta de experiencia e inseguridad por parte de los proveedores
- Puede no haber mucho entendimiento entre el equipo de la dependencia de TI de la organización contratante y los proveedores de servicios

En conclusión, el outsourcing de TI es un tema que tiene un nivel de madurez estable en muchas organizaciones y es una práctica muy aplicada hoy en día.

#### **4.1.8.4. Tipos de Outsourcing de TI**

[FGR98], presenta la siguiente tipología de outsourcing tecnológico:

*Tecnológico:* son aplicaciones que tienen que ver con las siguientes funciones: desarrollo y mantenimiento de software, incluidas todas sus fases como análisis, diseño, codificación, pruebas, mantenimiento y capacitación. Es en este punto donde se observa realmente la ventaja de reducir costos, porque si la empresa contratante asumiera esta responsabilidad, tendría que hacer una gran inversión. Otra función es la de mantenimiento y soporte de la infraestructura y administración de TI: el objetivo es subcontratar la administración de los centros de cómputo y todos los equipos; la ventaja de esto es, tener las herramientas y el software actualizados y disponibles. Otros de los servicios que caen dentro de esta tipificación y que las empresas contratan con mas frecuencia son los servicios de Help Desk, Call Center y de seguridad informática.

*Operativo:* en esta categoría se tienen en cuenta los procesos estratégicos del negocio; aquí hay funciones de desarrollo e investigación de nuevos productos, procedimientos para apoyar la gestión logística, la gestión financiera, la auditoría y el control, y las actividades de revisión y control de las operaciones.

*De servicios generales:* en esta categoría se incluyen todas aquellas operaciones que tienen que ver con tareas que permiten llevar a cabo los procesos del negocio y el buen desempeño de los mismos; aquí se llevan a cabo funciones de soporte y mantenimiento de equipos, mantenimiento a la red, instalaciones, configuración de equipos y servicios para ayudar a los usuarios de los puestos de trabajo de toda la organización.

*Estratégico:* en esta categoría se encuentran todas aquellas funciones que tienen que ver con la planeación y visión de la organización; las aplicaciones

para este tipo de outsourcing son: planeación estratégica, un tercero o asesor orienta a la compañía en la formulación de planes estratégicos, la idea es aprender metodologías de trabajo, otro tipo de aplicación puede ser capacitación del personal directamente implicado.

## **4.2. LA CALIDAD**

La calidad es un concepto que se ha puesto de moda durante las últimas décadas, más aún en el contexto del desarrollo de software. Es muy común estar desempeñando las labores cotidianas y escuchar hacer referencia a ese término, pero, ¿qué es calidad? La calidad es tal vez uno de los términos mas referenciados a nivel organizacional y a la vez más incomprendido. Son muchas las definiciones, los trabajos, las estrategias, los modelos, los estándares, las metodologías y los marcos de referencia que se han desarrollado en torno al tema de la calidad. Esta sección tiene como objetivo principal lograr un acercamiento a lo que realmente significa la calidad, entenderla bajo el contexto del desarrollo de software, y conceptualizar el aseguramiento de la calidad.

### **4.2.1. Definición**

Son muchas y variadas las definiciones que se han desarrollado en torno al tema de la calidad. Todas ellas definen perspectivas distintas bajo las cuales se analiza el mismo concepto y sugieren algunas claves que deben tomarse en cuenta para alcanzarla:

“La calidad es el agregado tangible e intangible de la funcionalidad, utilización, fiabilidad, ejecución, escala, apoyo, seguridad y cualquier otro factor considerado importante por el cliente y los objetivos comerciales de la organización.” [GEO04].

“La calidad consiste en no tener deficiencias.” [JMJ90].

“Grado en que un conjunto de características inherentes cumple con unos requisitos”<sup>10</sup>.

“Calidad (significa) conformidad con los requisitos.” [CRO87].

"Calidad es la medida en que un producto específico se ajusta a un diseño o especificación." [GIL74].

"Conjunto de esfuerzos efectivos de los diferentes grupos de una organización para la integración del desarrollo, del mantenimiento y de la superación de la calidad de un producto, con el fin de hacer posible la fabricación y servicio a satisfacción completa del consumidor y al nivel más económico" [FEI91].

"Calidad es satisfacer las expectativas del cliente. El Proceso de Mejora de la Calidad es un conjunto de principios, políticas, estructuras de apoyo y prácticas destinadas a mejorar continuamente la eficiencia y la eficacia de nuestro estilo de vida." [ATT56].

Entre este mar de definiciones, es de preferencia del autor la que se describirá subsecuentemente ya tal vez es la que mejor se acerca al concepto de calidad: “La Calidad es (1) un proceso bien definido, (2) para la fabricación de un producto útil, (3) que tiene valor agregado para ambos, él usuario final y él fabricante...” [LOR04]. La anterior definición establece tres aspectos clave en cuanto al tema de la calidad que serán tratados a continuación:

- *Un Proceso Bien Definido*: rara vez, un producto que se desarrolle de manera artesanal, sin seguir unas pautas o indicaciones mínimas, puede alcanzar las especificaciones que hayan sido establecidas. Toda actividad o tarea que se lleve a cabo requiere de un plan o unas guías de acción que permitan hacer lo que se tiene que hacer. La

---

<sup>10</sup> [ISO00], p. 8.

improvisación en la elaboración de productos puede dejar secuelas imborrables en los mismos, y por esto se requiere de la definición de procesos que guíen y que establezcan las pautas mínimas que deben satisfacerse para que los productos desarrollados cumplan con las especificaciones requeridas.

- *Fabricación de un Producto Útil:* los productos desarrollados deben ser de utilidad para sus usuarios. Los productos no deben surgir de la nada; deben ser la consecuencia de un mínimo análisis de las necesidades particulares de una organización o persona. Cada producto debe tener un fin, un objetivo, una utilidad para quién lo requiere; debe satisfacer la necesidad de algo o de alguien. De nada sirve un producto sin defectos o estéticamente aventajado, si no resuelve alguna necesidad o presenta alguna utilidad.
- *Tiene valor agregado para ambos, el usuario final y el fabricante:* la atención del fabricante a los pequeños detalles que van mas allá de los requisitos básicos definidos para el producto, son un factor diferenciador del mismo. Para desplazarse del sitio A al sitio B, se puede utilizar un Renault 4 o un Mercedes Benz, pero este último ofrece mayores y mejores prestaciones al usuario. Esto es lo que se denomina valor agregado, en pro del cual las organizaciones modernas de manera proactiva adelantan labores de innovación sobre sus productos y servicios que les permitan establecer brechas con sus competidores y así obtener una ventaja competitiva [RES06].

#### **4.2.2. Calidad del Software**

La Calidad del Software (CS, para abreviar) es una más de las disciplinas de la Ingeniería de Software y puede definirse como “la concordancia con los requerimientos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo documentados y con las características implícitas

que se esperan de todo software desarrollado profesionalmente” [PRE02]. Como es posible deducir de la definición anterior, la Calidad del Software define como elemento principal los requerimientos funcionales, ya que es sobre ellos donde se ejercen las distintas labores de valoración de la calidad. Debe hacerse énfasis, en que los requerimientos deben ser explícitamente establecidos, es decir, alguien de manera formal solicitó que se incluyeran ciertas características sobre el producto. Puede decirse, que si un producto software no satisface sus requerimientos funcionales o de rendimiento, es un producto que no satisface los atributos de calidad.

Otro elemento importante en la definición, se refiere a los estándares de desarrollo. Los estándares, las metodologías, los marcos de procesos y similares, establecen las pautas bajo las cuales un producto software debe ser desarrollado; estos elementos sirven de guía y muestran el camino que la organización desarrolladora debe seguir para implementar una solución informática aplicando la Ingeniería de Software. Debe tenerse en cuenta que si en un proyecto de desarrollo no se ciñe a ninguna metodología, siempre habrá falta de calidad; la utilización de este tipo de herramientas tiene como único fin producir software de alta calidad [SCA06]. Un factor importante que toma parte en este punto, es el Plan de Calidad. El Plan de Calidad toma como base los estándares y los procedimientos específicos que se hayan establecido y se constituye como el instrumento primordial para garantizar la calidad de los productos. Debe tenerse en cuenta que las especificaciones desarrolladas dentro del Plan de Calidad y los procedimientos establecidos, pueden variar de acuerdo a las circunstancias particulares de cada organización.

Finalmente, debe subrayarse la última parte de la definición que dicta que un producto software debe poseer ciertas características implícitas que se espera que todo producto desarrollado profesionalmente tenga. Estas características rara vez son solicitadas de manera formal por los clientes y/o los usuarios porque “se supone” que el producto software las debe poseer, por ejemplo, un usuario nunca pide que el producto sea “funcional” porque este tiene que serlo

para poder ser de utilidad. Estos son elementos obvios que los desarrolladores no deben dejar pasar sin ninguna explicación y que no son frecuentemente mencionados hasta que, por su ausencia, se pone en entredicho la calidad del producto.

#### **4.2.3. Calidad Producto y Calidad Proceso**

Cuando se habla de calidad una distinción importante que debe hacerse; una cosa es la calidad del producto la cual hace referencia a una serie de características que se espera que el producto software posea, es decir, se preocupa por el producto como producto final y no como este haya sido desarrollado. Otra es la que tiene que ver con la calidad del proceso, cuya principal preocupación es, que se sigan determinadas pautas, indicaciones, actividades o que satisfagan ciertas métricas en el proceso de desarrollo del producto. ¿Cómo se relacionan? Tanto la calidad del producto como la calidad del proceso se encuentran fuertemente relacionadas ya que los objetivos que se establezcan para la calidad del producto entrarían a determinar los objetivos que el proceso de su desarrollo debe satisfacer, es decir, la calidad del proceso siempre va estar en función de la calidad del producto.

Es importante tener en cuenta que el tema de la calidad en el contexto del software, es significativamente diferente al tema de la calidad en otro tipo de productos. Los productos software poseen características especiales que los convierten en un caso particular. Estas son [SCA06]:

- El software es un producto mental, no restringido por las leyes de la Física o por los límites de los procesos de fabricación. Es algo abstracto, y su calidad también lo es.
- Se desarrolla, no se fabrica. El coste está fundamentalmente en el proceso de diseño, no en la producción. Y los errores se introducen también en el diseño, no en la producción.

- El software no se deteriora con el tiempo. No es susceptible a los efectos del entorno, y su curva de fallos es muy diferente a la del hardware. Todos los problemas que surjan durante el mantenimiento estaban desde el principio, y afectan a todas las copias del mismo; no se generan nuevos errores.
- Es artesanal en gran medida. El software, en su mayoría, se construye a medida, en vez de ser construido ensamblando componentes existentes y ya probados, lo que dificulta aún más el control de su calidad. Aunque se ha escrito mucho sobre la reutilización del software, hasta ahora se han conseguido pocos éxitos tangibles.
- El mantenimiento del software es mucho más complejo que el mantenimiento del hardware. Cuando un componente de hardware se deteriora se sustituye por una pieza de repuesto, pero cada fallo en el software implica un error en el diseño o en el proceso mediante el cual se tradujo el diseño en código de máquina ejecutable.
- Es engañosamente fácil realizar cambios sobre un software, pero los efectos de estos cambios se pueden propagar de forma explosiva e incontrolada.
- Como disciplina, el desarrollo de software es aún muy joven, por lo que las técnicas de las que disponemos aún no son totalmente efectivas o no están totalmente calibradas.
- El software con errores no se rechaza. Se asume que es inevitable que el software presente errores.

#### **4.2.4. Aseguramiento de la Calidad**

El Aseguramiento de la Calidad es definido como “el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza en que el producto software satisfecerá los requisitos dados de calidad” [CUE99]. El Aseguramiento de la Calidad del Software (ACS, para abreviar) es una de las disciplinas que mas fuerza ha tomado en los últimos años ya que es el mecanismo que mejor apoya el tema de calidad bajo el contexto del desarrollo del software. Básicamente, el ACS puede entenderse como un enfoque sistemático y planeado para la evaluación de la calidad de los productos

software y su adherencia a los estándares, procesos y procedimientos establecidos [ATL06].

El ACS incluye el proceso de asegurar que los estándares y procedimientos se encuentran establecidos y que son seguidos con rigurosidad a través de todo el ciclo de vida de la adquisición del producto software. El cumplimiento de las especificaciones descritas en los estándares y procedimientos es determinado a través del monitoreo de los procesos, la evaluación del producto y las auditorías. Es importante tener en cuenta que, para desarrollar una estrategia seria y efectiva de aseguramiento de la calidad, los procesos a seguir en el ciclo de vida del desarrollo del producto deben incluir puntos de aprobación de aseguramiento de la calidad, es decir, la verificación que las especificaciones descritas en los estándares y los procedimientos, realmente deben estar siendo materializadas en el producto.

#### **4.2.5. Factores de Calidad**

Ya entrados en el mundo de la Calidad del Software, cabe hacerse la siguiente pregunta: ¿cómo se mide o se estima la calidad de un producto software? Distintos autores y organizaciones han desarrollado diversas técnicas y estrategias para dar solución a esta pregunta. Una de ellas, y con gran reconocimiento, es el llamado modelo de McCall. El modelo de McCall desarrolla el concepto de factores de calidad que pueden entenderse como atributos del producto software que pueden ser percibidos bajo la visión del usuario [CAM78]. Este modelo agrupa los factores de calidad en tres ejes, desde los cuales un usuario puede percibir la calidad del producto software. Cada uno de estos tres ejes tipifica los factores de calidad los cuales, a su vez, se desglosan en otros criterios. A continuación se describen brevemente los elementos del modelo de McCall.

#### 4.2.5.1. Eje 1: Operación del Producto

Este eje hace referencia a las características operativas del producto software. La tabla 4.1 describe los factores y los criterios que pertenecen a la Operación del Producto.

Factor	Criterios
<p><i>Facilidad de uso: el esfuerzo requerido para aprender el manejo de una aplicación, trabajar con ella, introducir datos y conseguir resultados.</i></p>	<ul style="list-style-type: none"> <li>• Facilidad de operación: Atributos del software que determinan la facilidad de operación del software.</li> <li>• Facilidad de comunicación: Atributos del software que proporcionan entradas y salidas fácilmente asimilables.</li> <li>• Facilidad de aprendizaje: Atributos del software que facilitan la familiarización inicial del usuario con el software y la transición del modo actual de operación.</li> <li>• Formación: El grado en que el software ayuda para permitir que nuevos usuarios apliquen el sistema.</li> </ul>
<p><i>Integridad: el grado con que puede controlarse el acceso al software o a los datos a personal no autorizado.</i></p>	<ul style="list-style-type: none"> <li>• Control de accesos. Atributos del software que proporcionan control de acceso al software y los datos que maneja.</li> <li>• Facilidad de auditoría: Atributos del software que facilitan la auditoría de los accesos al software.</li> <li>• Seguridad: La disponibilidad de mecanismos que controlen o protejan los programas o los datos.</li> </ul>
<p><i>Corrección: el grado en que una aplicación satisface sus especificaciones y consigue los objetivos encomendados por el cliente.</i></p>	<ul style="list-style-type: none"> <li>• Completitud: Atributos del software que proporcionan la implementación completa de todas las funciones requeridas.</li> <li>• Consistencia: Atributos del software que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación.</li> <li>• Trazabilidad o rastreabilidad: Atributos del software que proporcionan una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto.</li> </ul>

Tabla 4.1. Factores y criterios de la Operación del Producto. Adaptado de [CER06].

Factor	Criterios
<p><i>Fiabilidad: el grado que se puede esperar de una aplicación lleve a cabo las operaciones especificadas y con la precisión requerida.</i></p>	<ul style="list-style-type: none"> <li>• Precisión: Atributos del software que proporcionan el grado de precisión requerido en los cálculos y los resultados.</li> <li>• Consistencia.</li> <li>• Tolerancia a fallos: Atributos del software que posibilitan la continuidad del funcionamiento bajo condiciones no usuales.</li> <li>• Modularidad: Atributos del software que proporcionan una estructura de módulos altamente independientes.</li> <li>• Simplicidad: Atributos del software que posibilitan la implementación de funciones de la forma más comprensible posible.</li> <li>• Exactitud: La precisión de los cálculos y del control.</li> </ul>
<p><i>Eficiencia: la cantidad de recursos hardware y software que necesita una aplicación para realizar las operaciones con los tiempos de respuesta adecuados.</i></p>	<ul style="list-style-type: none"> <li>• Eficiencia en ejecución: Atributos del software que minimizan el tiempo de procesamiento.</li> <li>• Eficiencia en almacenamiento: Atributos del software que minimizan el espacio de almacenamiento necesario.</li> </ul>

Tabla 4.1. Factores y criterios de la Operación del Producto. Adaptado de [CER06].

#### 4.2.5.2. Eje 2: Revisión de Producto

La revisión del producto hace referencia a la capacidad del software para soportar cambios. La tabla 4.2 describe las características de este eje.

Factor	Criterios
<p><i>Facilidad de mantenimiento: el esfuerzo requerido para localizar y reparar errores.</i></p>	<ul style="list-style-type: none"> <li>• Modularidad.</li> <li>• Simplicidad.</li> <li>• Consistencia.</li> <li>• Concisión: Atributos del software que posibilitan la implementación de una función con la menor cantidad de códigos posible.</li> <li>• Auto descripción: Atributos del software que proporcionan explicaciones sobre la implementación de las funciones.</li> </ul>

Tabla 4.2. Factores y criterios de la Revisión del Producto. Adaptado de [CER06].

Factor	Criterios
<p><i>Facilidad de prueba: el esfuerzo requerido para probar una aplicación de forma que cumpla con lo especificado en los requisitos.</i></p>	<ul style="list-style-type: none"> <li>• Modularidad.</li> <li>• Simplicidad.</li> <li>• Auto descripción.</li> <li>• Instrumentación: Atributos del software que posibilitan la observación del comportamiento del software durante su ejecución para facilitar las mediciones del uso o la identificación de errores.</li> </ul>
<p><i>Flexibilidad: el esfuerzo requerido para modificar una aplicación en funcionamiento.</i></p>	<ul style="list-style-type: none"> <li>• Auto descripción.</li> <li>• Capacidad de expansión: Atributos del software que posibilitan la expansión del software en cuanto a capacidades funcionales y datos.</li> <li>• Generalidad: Atributos del software que proporcionan amplitud a las funciones implementadas.</li> <li>• Modularidad.</li> </ul>

Tabla 4.2. Factores y criterios de la Revisión del Producto. Adaptado de [CER06].

#### 4.2.5.3. Eje 3: Transición del Producto

La transición del producto hace referencia a la capacidad que tiene el software para adaptarse a nuevos entornos. La tabla 4.3 describe los factores de calidad y los criterios que pertenecen a este eje.

Factor	Criterios
<p><i>Reusabilidad: Grado en que partes de una aplicación pueden utilizarse en otras aplicaciones</i></p>	<ul style="list-style-type: none"> <li>• Auto descripción.</li> <li>• Generalidad.</li> <li>• Modularidad.</li> <li>• Independencia entre sistema y software: Atributos del software que determinan su dependencia del entorno operativo.</li> <li>• Independencia del hardware: Atributos del software que determinan su dependencia del hardware.</li> </ul>

Tabla 4.3. Factores y criterios de la Transición del Producto. Adaptado de [CER06].

Factor	Criterios
<p><i>Interoperabilidad: El esfuerzo necesario para comunicar la aplicación con otras aplicaciones o sistemas informáticos</i></p>	<ul style="list-style-type: none"> <li>• Modularidad.</li> <li>• Compatibilidad de comunicaciones: Atributos del software que posibilitan el uso de protocolos de comunicación e interfaces estándar.</li> <li>• Compatibilidad de datos: Atributos del software que posibilitan el uso representaciones de datos estándar.</li> <li>• Estandarización en los datos: El uso de estructuras de datos y de tipos estándar a lo largo de todo el programa.</li> </ul>
<p><i>Portabilidad: El esfuerzo requerido para transferir la aplicación a otro hardware o sistema operativo</i></p>	<ul style="list-style-type: none"> <li>• Auto descripción.</li> <li>• Modularidad.</li> <li>• Independencia entre sistema y software.</li> <li>• Independencia del hardware.</li> </ul>

Tabla 4.3. Factores y criterios de la Transición del Producto. Adaptado de [CER06].

Es importante anotar, que con el establecimiento de estos factores de calidad es posible evaluar la calidad de un producto software. Básicamente, se trata de ponderar cada uno de estos factores de calidad y establecer una escala numérica para calificar los criterios que definen. Luego de efectuar la calificación y promediarla a cada factor, se efectúa la ponderación y se suman los resultados. La ponderación que se escoja depende únicamente de la importancia que tenga cada factor respecto al producto software y de las necesidades y requerimientos que se hayan establecido.

Por último, y como parte fundamental del tema de la Calidad del Software, se encuentran los Modelos de Calidad. Los Modelos de Calidad apuntan al desarrollo y establecimiento de un conjunto de propiedades de calidad deseables en un producto software. Por lo general (o al menos en los estándares modernos), este objetivo es conseguido mediante el desarrollo y acatamiento juicioso de una serie de procesos que han probado ser efectivos. Los Modelos de Calidad son el tema a tratar en las secciones subsiguientes.

### **4.3. MODELOS Y ESTÁNDARES DE CALIDAD**

#### **4.3.1. Norma ISO 9000**

La norma ISO 9000 representa una familia de estándares que son utilizados para establecer sistemas de calidad. Diseñados para facilitar la gestión y el aseguramiento de la calidad, estos estándares son verdaderos estándares internacionales en el sentido estricto de la palabra. Esta internacionalidad es una verdadera ventaja en cuanto al tema de los mercados internacionales se refiere, ya que algunos países usaban estándares locales para proteger las industrias nacionales. Con la ISO 9000, una certificación que se obtenga en Colombia es igualmente válida en cualquier país del mundo, lo cual es sumamente favorable para a aquellas organizaciones que conforman la industria del software y que pretenden alcanzar mercados internacionales [UNO03].

La norma ISO 9000 es aplicable a cualquier tipo de industria; puede ser utilizada tanto en los sectores manufactureros, comerciales y de servicios, y goza de amplia difusión y conocimiento. Además de lo anterior, la norma se caracteriza por tener un gran alcance y flexibilidad a la hora de implementarla en distintos tipos de organizaciones, dado que establece los requerimientos que un sistema de calidad debe satisfacer, pero no prescribe acciones para alcanzarlos. Sin embargo, un estándar con estas características debe ser muy genérico para poder ser compatible con todos los sectores y tipos de organizaciones y, por lo tanto, una orientación detallada sobre calidad en áreas específicas de interés no es posible. Adicionalmente, un estándar tan ampliamente definido como la ISO 9000, requiere de análisis y mucha interpretación antes que pueda ser adoptado.

#### **4.3.1.1. Reseña Histórica**

Los estándares que conforman la norma ISO 9000 fueron publicados por primera vez en 1987, y desde ese entonces han sufrido una serie de cambios como consecuencia de las revisiones que sobre ella se realizan que, como compromiso de la ISO [SAN03], se llevan a cabo con una periodicidad de cinco años.

La primera revisión del estándar se efectuó en 1994, en la que se incluyeron un conjunto mínimo de requisitos para el establecimiento y mantenimiento del sistema de gestión y aseguramiento de la calidad para la Ingeniería de Software. En este sistema se identificaban 20 áreas principales a trabajar y 177 indicadores de gestión [VRR06]. Posteriormente, y con base en retroalimentación obtenida de muchos usuarios y de la crítica especializada, la ISO 9000 fue actualizada nuevamente en el año 2000. En síntesis, las mejoras que se introdujeron son:

- Se emplea una aproximación de gestión basada en el proceso
- Se hace compatible con otros estándares (p.e., ISO 14000)
- Establece requisitos para la mejora continua del sistema de calidad
- Se aproxima a las necesidades de los participantes que nada tienen que ver con el sistema de calidad (p.e., clientes, proveedores)
- Se hace más usable por clientes y usuarios

Con las mejoras introducidas, la ISO 9000 pasa de ser un estándar prescriptivo a ser un estándar más adecuado a las necesidades de las organizaciones modernas, basándose en procesos que tienen como objetivos fundamentales la satisfacción de los clientes, la calidad de los productos y la mejora continua.

#### **4.3.1.2. La Familia ISO 9000**

El término ISO 9000 [SUI06] hace referencia a una familia de estándares. En la actualidad, esta familia incluye tres estándares de calidad que son comúnmente referenciados como “la ISO 9000”: ISO 9000:2000, ISO 9001:2000 e ISO 9004:2000. La ISO 9001:2000 presenta los requerimientos, mientras que la ISO 9000:2000 y la ISO 9004:2000 proporcionan orientación en cuanto a la implementación de los procesos. En términos más formales estos estándares son:

- ISO 9000. Fundamentos y vocabulario
- ISO 9001. Requisitos para aseguramiento de la calidad
- ISO 9004. Directrices para la mejora del rendimiento

Es importante tener en cuenta que estos estándares se orientan a procesos y no a productos como mucha gente podría pensar. Debe considerarse además, que la ISO 9001 y la ISO 9004 son estándares complementarios. Mientras que el primero es usualmente empleado para propósitos de certificación y se concentra en la eficacia del sistema de gestión de calidad para dar cumplimiento a los requerimientos del cliente, la ISO 9004 es empleada por organizaciones que buscan la mejora continua de sus procesos.

##### **4.3.1.2.1. ISO 9001:2000**

El estándar más importante dentro de la familia de estándares ISO 9000 es el estándar ISO 9001. La razón para hacer la anterior afirmación es que la ISO 9001 es la única norma dentro de la familia a través de la cual una organización puede obtener una certificación. En palabras simples, cumpliendo con los requerimientos establecidos en la norma una organización puede ser certificada, sin importar si se satisfacen o no los criterios establecidos dentro de

las otras normas que componen la ISO 9000. Respecto a este tema, [SUI06] comenta:

Este es el estándar de requerimientos usado para valorar la capacidad de una organización para reunir los requerimientos del cliente y los requerimientos regulatorios aplicables, y de ese modo alcanzar la satisfacción del cliente. En este momento es el único estándar en la familia ISO 9000 a través del cual se brinda una fuente de certificación.

Pese a lo anterior, debe tenerse en cuenta que el reporte técnico del estándar ISO 9001:2000 no tiene más de treinta páginas, y como se mencionó anteriormente la amplia cobertura que pretende la norma hace que las instrucciones allí contenidas sean muy generales. Por dicho motivo, para lograr entender realmente la norma y para facilitar el montaje de un sistema de gestión de la calidad, es recomendable recurrir a las indicaciones contenidas en la ISO 9000:2000 y en la ISO 9004:2000.

#### **4.3.1.2.2. ISO 9000:2000**

La ISO 9000:2000 explica las bases sobre las cuales se desarrolló la norma ISO 9000 y ofrece un glosario donde se explica toda la terminología utilizada dentro del estándar. En palabras de [SUI06] es:

Establece un punto de inicio para entender los estándares y define los términos fundamentales y definiciones usadas en la familia de estándares ISO 9000. Es importante evitar malos entendidos en cuanto a su uso.

#### **4.3.1.2.3. ISO 9000-3:1997**

Como menciona [UNO03], el software tiene sus propios atributos: es más complejo que el hardware, es inmaterial, no tiene límites físicos, y es fácil de cambiar pero las consecuencias de dichos cambios son difíciles de evaluar. Tomando lo anterior en consideración y debido a que los productos software difieren por naturaleza de los productos desarrollados en el resto de sectores productivos, la ISO desarrolló una serie de pautas que aplican de forma exclusiva al tema del desarrollo de software. Estas pautas se encuentran contenidas en la parte tres de la norma ISO 9000 y especifican como aplicar los requerimientos de la ISO 9001 a los productos software. La ISO 9000-3:1997 todavía toma como referencia los requisitos establecidos en la norma ISO 9001:1994, por lo tanto no se encuentra actualizada y no se ajusta a las necesidades recientes del sector software.

#### **4.3.1.2.4. ISO 9004:2000**

La ISO 9004 proporciona directrices que van mas allá del alcance establecido en la ISO 9001 y que buscan establecer el potencial de mejora del desempeño de la organización. Con este propósito, la ISO 9004 toma en consideración la eficacia y eficiencia relativas al sistema de gestión de la calidad y promueve el mejoramiento continuo del mismo. En concreto, si se compara la ISO 9001 con la ISO 9004 puede observarse que los objetivos relativos a la satisfacción del cliente y a la calidad del producto, son extendidos para incluir aspectos tales como el desempeño organizacional y la satisfacción de las partes interesadas. [SUI06] declara al respecto:

Este manual estándar provee indicaciones para el mejoramiento continuo de el sistema de gestión de la calidad para beneficiar a todas la partes a través del sostenimiento de la satisfacción del cliente.

### 4.3.1.3. Contenido de la ISO 9001

En su versión 2000, la ISO 9001 abandona la estructura de veinte secciones contenida en el estándar de 1994 y propone una comprendida por cinco secciones. Estas secciones reflejan el ciclo de Deming de Planear, Hacer, Verificar y Actuar [ITS01] y la relación entre ellas puede verse en la figura 4.4.

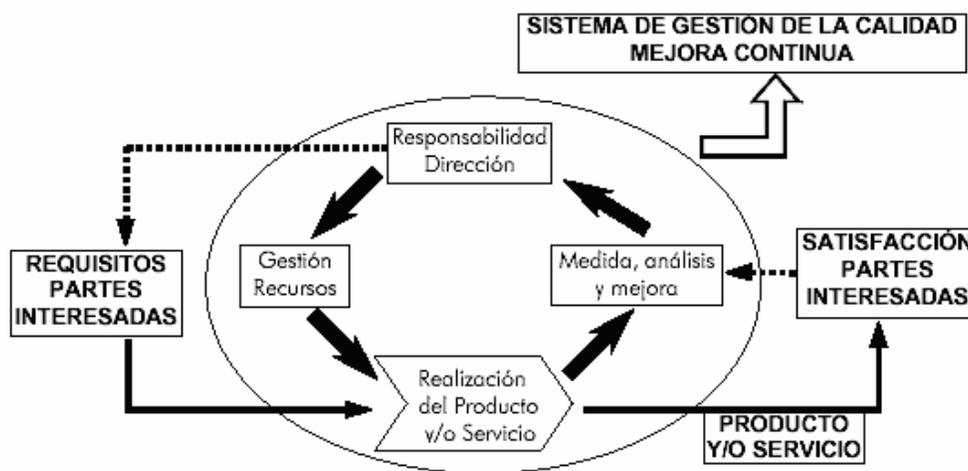


Figura 4.4. Relación entre secciones de la norma ISO 9001. Fuente [BUP06].

*Sección 4.* Sistema de Gestión de la Calidad: establece los requerimientos globales del sistema de gestión de la calidad, así como también las pautas para desarrollar la documentación y llevar los registros. Los requerimientos clave de esta sección son [ISU06]:

- Establecer, documentar, implementar, mantener y mejorar continuamente el sistema de gestión de la calidad
- Documentar las políticas, los objetivos y el manual de calidad
- Construir la documentación requerida por el estándar y por la misma organización en aras de planear, operar y controlar efectivamente los procesos
- Hacer gestión efectiva de la documentación (gestión de la configuración)
- Establecer y llevar registros

*Sección 5. Responsabilidades de la Dirección:* describe las responsabilidades de la alta dirección respecto al sistema de gestión de la calidad. Dichas responsabilidades hacen referencia a la verificación de los compromisos que se establezcan, el enfoque al cliente, la planeación, el establecimiento de objetivos, la revisión de los sistemas y el manejo de las comunicaciones internas. En concreto los requerimientos son [ISU06]:

- Proveer evidencia del compromiso de la dirección con el desarrollo, implementación y mejoramiento continuo de la eficiencia del sistema de gestión de la calidad
- Asegurar que los requerimientos del cliente son establecidos y satisfechos
- Establecer la política de calidad y asegurar que esta proporciona un marco de trabajo para establecer y revisar los objetivos de calidad
- Establecer objetivos de calidad a distintos niveles organizacionales, asociarlos a labores o funciones relevantes, y asegurar que dichos objetivos son medibles y que se ajustan a la política de calidad
- Desarrollar actividades de planeación para el sistema de gestión de la calidad
- Asegurar que las responsabilidades, el personal y su interrelación, se encuentran definidas y que han sido difundidas en la organización
- Delegar un representante de la dirección
- Asegurar que los procesos de comunicación dentro de la organización están establecidos y que estos son adecuados
- Desarrollar actividades de revisión y demostrar que las decisiones y acciones son tomadas respecto a actividades de mejoramiento

*Sección 6. Gestión de los Recursos:* esta sección hace referencia a los recursos requeridos por la organización para que el sistema de calidad sea

viable y para llenar las expectativas del cliente. Además del tema de recursos humanos y necesidades de entrenamiento, también cubre el tema de los recursos físicos empleados en el proceso. Los requerimientos relativos a esta sección son [ISU06]:

- Asegurar la competencia, concientización y entrenamiento de los empleados. Evaluar la efectividad de las acciones tomadas
- Asegurar que el personal es consciente de la importancia de sus actividades y de cómo estas contribuyen al logro de los objetivos de calidad
- Llevar registros del nivel de educación, habilidades y experiencia del personal
- Identificar, proporcionar y mantener los recursos necesarios para lograr un producto conforme<sup>11</sup>, incluyendo los servicios de soporte que implican actividades como el transporte, comunicación y programas de mantenimiento
- Identificar y mantener los factores del entorno requeridos para desarrollar un producto conforme

*Sección 7. Realización del Producto:* establece los requerimientos que los productos y servicios deben reunir y se refiere a los procesos implicados en el desarrollo de los mismos. Actividades tales como la revisión de contratos, la compra de insumos, y el diseño y desarrollo de productos, son tocados en esta sección. En concreto se pretende [ISU06]:

- Planear y desarrollar los procesos requeridos para la realización de productos
- Desarrollar procesos relacionados con el cliente, incluyendo aquellos que tienen que ver con la revisión de contratos y las comunicaciones con el mismo

---

<sup>11</sup> Según [UNG06], se dice que un producto es conforme cuando este cumple sus requisitos.

- Desarrollar requisitos para el diseño y desarrollo de productos e incluir control de cambios
- Desarrollar requisitos para la compra de provisiones
- Controlar los procesos de producción y prestación de servicios
- Identificación y trazabilidad
- Controlar la propiedad del cliente
- Preservar el producto
- Controlar los dispositivos de monitoreo y medición

*Sección 8. Medición, Análisis y Mejoramiento.* Establece los requerimientos para el desarrollo de las actividades de medición, lo cual incluye las métricas para calcular la satisfacción del cliente, el análisis de datos y el mejoramiento continuo. Los requerimientos de esta sección son [ISU06]:

- Planear e implementar el monitoreo, medición, análisis y mejoramiento continuo de los procesos
- Monitorear la información relativa al usuario como también los indicadores de desempeño del sistema de gestión de la calidad
- Conducir auditorías internas
- Monitorear y medir los procesos
- Monitorear y medir los productos
- Controlar los productos no conformes
- Analizar datos
- Facilitar el mejoramiento continuo
- Implementar acciones correctivas
- Implementar acciones preventivas

#### **4.3.1.4. Los Ochos Principios**

En términos generales, la versión 2000 de la ISO 9000 es mucho más orientada al cliente y al mejoramiento continuo que la versión de 1994. Acorde con la ISO, los estándares que componen la familia están fundamentados en ocho principios de calidad que tienen como objetivo principal el incremento del desempeño organizacional y la satisfacción del cliente. Una pregunta que tiene cabida en este punto es, ¿cómo una organización puede estar segura que cumple con estos principios? Según [ISP06], si una organización implementa un sistema de calidad que reúna los requerimientos establecidos en la norma ISO 9001:2000, automáticamente dicha organización estará cumpliendo con los principios de calidad.

##### **4.3.1.4.1. Organización Centrada en el Cliente**

Este principio se fundamenta en la idea que las organizaciones dependen y subsisten de sus clientes y, por tanto, deben siempre procurar llenar sus expectativas. Adicionalmente, se plantea que la organización debe ser proactiva en la identificación temprana de las necesidades futuras del cliente.

##### **4.3.1.4.2. Liderazgo**

Este principio resalta la importancia y los beneficios que la capacidad de liderazgo puede traer a una organización. Básicamente, se promueve la idea que los líderes, haciendo uso de su poder de convocatoria y de convicción, estimulan al personal a que hagan parte activa del proyecto organizacional. Lo anterior, genera un entorno ideal para que los objetivos organizacionales sean alcanzados.

#### 4.3.1.4.3. Involucrar a la Gente

Este principio promueve la idea de motivar a las personas e involucrarlas de lleno en el proyecto organizacional, lo que hace que la organización pueda explotar o sacar máximo provecho de sus habilidades.

#### 4.3.1.4.4. Enfoque hacia Procesos

Básicamente, se plantea la idea que utilizando un enfoque hacia procesos, una organización puede obtener mejores resultados y ser más eficiente en el desarrollo de sus actividades. De igual modo, los recursos que consumen estas actividades pueden ser optimizados.

#### 4.3.1.4.5. Enfoque Sistémico de la Gestión

Este principio resalta los beneficios de manejar procesos interrelacionados bajo un enfoque sistémico. Básicamente, se sostiene que los objetivos organizacionales pueden ser alcanzados más fácilmente, si sus procesos son entendidos y gestionados como un sistema.

#### 4.3.1.4.6. Mejora Continua

Este es uno de los principios más importantes que define la norma y plantea que un objetivo permanente de cualquier organización debe ser el mejoramiento continuo del desempeño organizacional. Lo anterior otorga a las organizaciones la capacidad de reaccionar rápido ante las oportunidades del mercado y pone en sintonía las actividades de mejoramiento con la estrategia organizacional.

#### 4.3.1.4.7. Enfoque Objetivo para la Toma de Decisiones

Este principio promulga el análisis de datos y de la información como estrategia fundamental para la toma efectiva de decisiones. A través de la aplicación de un enfoque objetivo para la toma de decisiones, una organización puede revisar, cuestionar, y cambiar planes de acción y decisiones. Adicionalmente, la organización adquiere la capacidad de valorar la efectividad de las decisiones tomadas en el pasado, comparándolas y haciendo uso de los registros fácticos presentes.

#### 4.3.1.4.8. Relaciones Mutuamente Beneficiosas

Este principio parte de la base que tanto organizaciones como proveedores son interdependientes, y que de una relación sana entre ellos depende que los dos puedan crear valor. Es importante tener en cuenta que cuando la relación entre organización y proveedor es beneficiosa para ambas partes, estas adquieren mayor capacidad de respuesta ante los cambios del mercado y ante las necesidades de los clientes, y pueden alcanzar la optimización de costos y de recursos.

### **4.3.2. Estándar ISO/IEC 15504**

En 1991 la ISO/IEC JTC1/SC7 inició un estudio motivado por la necesidad de crear un estándar para la valoración de procesos software. Tomando como argumentos los resultados de este estudio y con el fin de estandarizar y mejorar las metodologías para la valoración de software existentes en la época, en 1993 se creó el proyecto de Mejoramiento del Proceso Software y Determinación de la Capacidad (SPICE, por sus siglas en inglés), del cual también hicieron parte la IEEE y el SEI.

El primer logro de este proyecto fue alcanzado en 1995 con la liberación del primer borrador del estándar para la valoración del proceso software, el cual es más comúnmente referenciado como la suite de documentos SPICE (o SPICE versión 1). Estos documentos son públicos y se encuentran disponibles en Internet en el sitio <http://www.sqi.gu.edu.au/spice/>.

Posteriormente y bajo la conducción del grupo de trabajo para la valoración de procesos (ISO/IEC JTC1/SC7/WG10), la suite de documentos SPICE fue convertida en reporte técnico, liberada en 1998 y denominada como ISO/IEC (TR) 15504, estándar para la valoración del proceso software. Desde la fecha de liberación hasta hoy, un estimado de 3000 valoraciones de procesos software han sido realizadas alrededor del mundo utilizando el estándar. Tomando como base la retroalimentación obtenida de la comunidad internacional, el estándar está siendo revisado y se espera que una versión estable y completa sea publicada próximamente [ISS06].

#### **4.3.2.1. La Necesidad de Otro Estándar**

Dada la creciente población de estándares, marcos de referencia, metodologías, modelos y similares, es comprensible cuestionarse acerca de la necesidad de otro estándar. A continuación se citan dos de las características más importantes que diferencian y agregan valor a la ISO/IEC 15504.

##### **4.3.2.1.1. Capaz de Realizar Comparaciones Realistas**

Uno de los beneficios más importantes de la ISO/IEC 15504 es su capacidad para comparar “manzanas con manzanas” [WIS06]. La ISO/IEC 15504 propone un conjunto mínimo de requerimientos que aseguran que los resultados de la valoración sean objetivos, imparciales, consistentes, repetibles y representativos de los procesos valorados. De manera conjunta, el estándar formula la valoración de la capacidad de los procesos software haciendo uso del framework de medición que propone, y los resultados obtenidos pueden ser

fácilmente comparados cuando se considera que los alcances de las evaluaciones realizadas son similares [ISO03].

#### 4.3.2.1.2. Evaluación de Procesos Individuales

La ISO/IEC 15504 establece el conjunto total de procesos involucrados en el proceso de desarrollo de software y permite evaluar individualmente cada uno de ellos. Dado que particularizando la valoración de los procesos se toma mucha más información en cuenta, es posible establecer un perfil organizacional (dependencia organizacional que administra el desarrollo de software o empresa desarrolladora de software) más acertado y se logran resultados más confiables y precisos. El hecho que la ISO/IEC 15504 permita hacer valoraciones individuales de los procesos involucrados en el desarrollo de software, como por el ejemplo el Gestión de la Configuración, es su característica y beneficio más importante.

#### 4.3.2.2. Visión General de la Norma ISO/IEC 15504

El marco para la valoración de procesos software entregado por la norma ISO/IEC 15504, puede ser utilizado por organizaciones implicadas en la planificación, gestión, control, monitoreo y mejora de los procesos de adquisición, desarrollo, transición, operación y soporte del producto software [ALG97].

En la figura 4.5 se muestran los componentes que hacen parte fundamental del estándar. Estos componentes y sus respectivas interacciones, resumen la estrategia propuesta por la ISO/IEC 15504 para obtener resultados de valor y sacar provecho de la misma. En su forma más básica, la evaluación del proceso consiste en examinar los procesos utilizados por una organización para determinar si son lo suficientemente efectivos para alcanzar los objetivos que definen. Los resultados obtenidos en la evaluación pueden ser utilizados

ya sea para medir la capacidad de un proceso en particular, la capacidad del proceso global de desarrollo, o para adelantar labores de mejoramiento sobre dichos procesos.

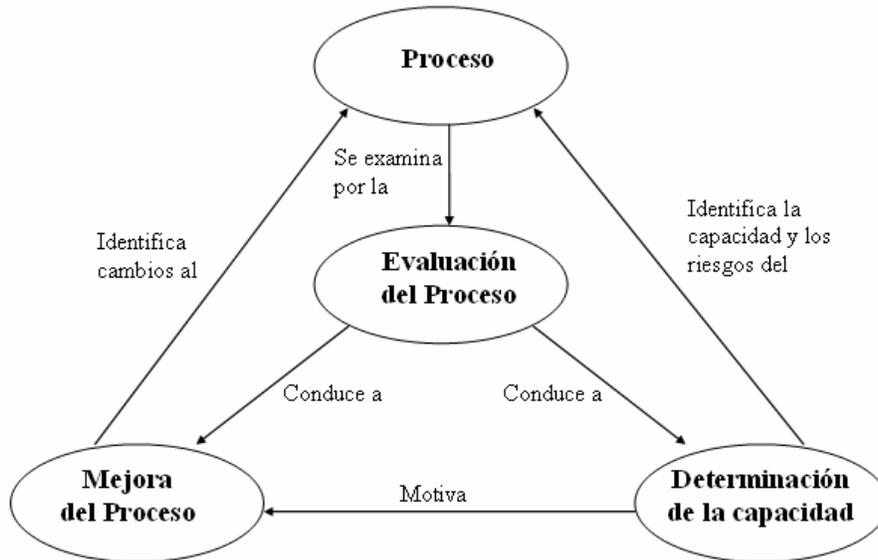


Figura 4.5. Estrategia propuesta por la ISO/IEC 15504 para obtener resultados. Fuente [ALG97].

#### 4.3.2.3. Centralización en la Valoración del Proceso

Tras haber analizado la retroalimentación recibida por parte de la comunidad internacional y luego de tres años de revisión, la versión completa del ya catalogado como estándar internacional ISO/IEC 15504 ha venido siendo publicada entre los años 2003 y 2006. La nueva versión cuyo título paso de “Software Process Assessment” a “Information Technology — Process Assessment” consta de 5 partes, lo cual es mucho menos que la versión anterior que constaba de una colección de 9 documentos que en total sumaban cerca de 600 páginas. Esta es una de las primeras grandes diferencias con la versión anterior. La segunda gran diferencia tiene que ver con el hecho que la nueva versión no se encuentra restringida a los procesos que tienen que ver con el ciclo de vida del software y, por el contrario, puede ser aplicada a cualquier tipo de procesos. Adicionalmente, la nueva versión de la ISO/IEC

15504 propone una fuerte orientación a la valoración de procesos, lo que se constituye como insumo principal para el mejoramiento de los mismos. El estándar consiste de los siguientes documentos [ISI05]:

- Parte 1. Conceptos y vocabulario
- Parte 2. Efectuando una valoración
- Parte 3. Guía sobre el desarrollo de una valoración
- Parte 4. Guía sobre el uso para el mejoramiento del proceso y para la determinación de la capacidad del proceso
- Parte 5. Un modelo ejemplar para la valoración del proceso

La parte 1 explica los conceptos generales y el vocabulario utilizado en el resto de documentos. La parte 2 contiene el material normativo considerado como obligatorio para estar en conformidad con el estándar. La parte 3 define los requerimientos para hacer valoraciones cuyas salidas sean repetibles, confiables y consistentes. Por último, las partes 4 y 5 contienen material informativo que sirve como orientación para la realización de valoraciones y para la interpretación y utilización de los resultados arrojados [DUS03].

#### **4.3.2.4. Modelo Externo de Referencia de Proceso**

La nueva versión del estándar ISO/IEC 15504 continúa siendo un modelo de dos dimensiones: la dimensión de proceso y la dimensión de capacidad. La diferencia radica en que la nueva versión externaliza la dimensión de proceso en un modelo de referencia de proceso. Lo anterior significa que la dimensión de proceso ya no hace parte del nuevo estándar y su modelo de referencia ha sido reubicado e incluido como un anexo de la norma ISO/IEC 12207. Este cambio se fundamenta en el hecho que el estándar no pretende hacer una prescripción de los procesos que una organización debería seguir sino, más bien, orientarse a la valoración de los mismos. La norma ISO/IEC 15504 es un marco de referencia que propone métodos para la valoración de procesos y no define explícitamente metodologías o modelos a seguir. Los modelos de

referencia de proceso han venido siendo desarrollados por fuera de los estándares ISO, tal vez con el fin de facilitarles la conformidad y cumplimiento de ciertos requerimientos. En el caso de la norma ISO/IEC 15504 estos requerimientos son descritos en la parte 2, la cual establece que con el fin que un modelo de referencia de proceso sea válido, éste debe describir los procesos en términos de propósitos y de resultados, al igual que las relaciones existentes entre los procesos definidos. Algunos modelos de referencia de proceso ya existen o están bajo desarrollo, como es el caso de la ya mencionada norma ISO/IEC 12207 AMD y de CMMI. Desde el punto de vista del proveedor de software, el estándar ISO/IEC 15504 no implica un cambio en el modelo de mejora del proceso software siempre y cuando éste siga usando un modelo que esté en concordancia.

#### **4.3.2.5. Marco de Referencia para la Medición**

La dimensión de capacidad del Modelo de Valoración de Proceso (PAM, por sus siglas en inglés) consiste de un marco de referencia de medición. Dicho marco de referencia define seis niveles de capacidad de proceso y sus respectivos atributos de proceso [ISS06].

##### **4.3.2.5.1. Niveles de Capacidad**

###### **4.3.2.5.1.1. Nivel 0: Incompleto**

No se tiene una idea clara del propósito del proceso. Es difícil identificar las salidas o productos de trabajo arrojados por el proceso. Los resultados del proceso son impredecibles.

#### 4.3.2.5.1.2. Nivel 1: Realizado

Por lo general se alcanza el propósito que el proceso define. Frecuentemente dicho propósito no es alcanzado ciñéndose rigurosamente al plan de trabajo definido. Existe un consenso entre el equipo de trabajo en que una acción debe ser ejecutada en el alcance y tiempos requeridos. Es posible identificar los productos que fueron arrojados por el proceso, lo cual prueba que se consiguió el propósito del mismo.

#### 4.3.2.5.1.3. Nivel 2: Gestionado

El proceso entrega productos de calidad aceptable dentro del plan de trabajo definido. El proceso es llevado a cabo con el plan definido. Los productos se apegan a estándares y requerimientos. La principal distinción de este nivel respecto al “Realizado” es que el desempeño del proceso es planeado y administrado.

#### 4.3.2.5.1.4. Nivel 3: Establecido

El proceso es llevado a cabo usando un proceso establecido y basado en los principios de la Ingeniería de Software. El proceso se adecua a las necesidades de cada proyecto y se encuentra estrictamente documentado. Los recursos necesarios para llevar a cabo el proceso se encuentran disponibles. La principal distinción de este nivel respecto al “Gestionado” es que el proceso establecido es planeado y administrado partiendo de un proceso estándar.

#### 4.3.2.5.1.5. Nivel 4: Predecible

El proceso definido es llevado a cabo de manera consistente en la práctica, con límites de control preestablecidos para alcanzar los objetivos propuestos.

Existen métricas detalladas del desempeño que son analizadas. Esto lleva a entender cuantitativamente la capacidad del proceso y a mejorar la habilidad de predecir su desempeño. El desempeño del proceso y la calidad de los productos de trabajo son manejados cuantitativamente. La principal distinción de este nivel respecto al “Establecido” es que el proceso es cuantitativamente entendido y controlado.

#### 4.3.2.5.1.6. Nivel 5: Optimizable

La ejecución del proceso es optimizada para localizar y satisfacer las necesidades actuales y futuras de la organización. Se establecen objetivos cuantitativos de eficiencia y efectividad en función de los objetivos de la organización. La optimización puede llevar a la investigación y adopción de ideas innovadoras o productos tecnológicos novedosos que incluyan y modifiquen el proceso definido.

#### 4.3.2.5.2. Atributos del Proceso

La medición de la capacidad del proceso esta basada en un conjunto de nueve atributos de proceso (PA, por sus siglas en ingles) los cuales se encuentran relacionados con los niveles de capacidad del 1 hasta el 5. La tabla 4.4 describe dichos atributos.

<b>Nivel</b>	<b>Atributo del Proceso</b>	<b>Descripción</b>
1	1.1. Rendimiento del Proceso	El proceso emplea un conjunto de prácticas, que son iniciadas por unos productos identificables y produce unos productos identificables, que satisfacen el propósito del proceso.
2	2.1. Gestión del Rendimiento	La ejecución del proceso se gestiona para producir productos en un plazo de tiempo y con unos requisitos preestablecidos.
2	2.2. Gestión del Producto	La ejecución del proceso se gestiona para producir productos que se documentan y se controlan satisfaciendo sus requisitos funcionales y no funcionales, de acuerdo con los objetivos de calidad del producto del proceso.

Tabla 4.4. Atributos del proceso(s) de desarrollo de software. Adaptado de [PRA06].

Nivel	Atributo del Proceso	Descripción
3	3.1. Definición del Proceso	La ejecución del proceso utiliza una definición de proceso basada en un proceso estándar, que permite contribuir a los objetivos de negocio definidos en la organización.
3	3.2. Recursos del Proceso	La ejecución del proceso utiliza eficazmente recursos humanos con las habilidades adecuadas y una infraestructura de proceso que contribuyen a los objetivos de negocio definidos de la organización.
4	4.1. Medición del Proceso	La ejecución del proceso se soporta por los objetivos y mediciones que son utilizadas para asegurar que la implementación del proceso contribuye a la consecución de los objetivos.
4	4.2. Control del Proceso	La ejecución del proceso se controla a través de la recopilación y análisis de mediciones para controlar y corregir, donde sea necesario, el rendimiento del proceso para lograr fiablemente los objetivos del proceso definidos.
5	5.1. Cambio de Proceso	Los cambios a la definición, gestión y rendimiento del proceso son controlados mejor para conseguir los objetivos de negocio de la organización.
5	5.2. Mejora Continua	Los cambios a los procesos se identifican y se implementan para asegurar la mejora continua en el cumplimiento de los objetivos del negocio definidos por la organización.

Tabla 4.4. Atributos del proceso(s) de desarrollo de software. Adaptado de [PRA06].

Los atributos de proceso son evaluados utilizando una escala porcentual que mide el grado en el cual dichos atributos han sido alcanzados. Posteriormente, dicho porcentaje es asociado a una representación cualitativa según la siguiente tabla:

Calificación	Porcentaje	Descripción
N. No Alcanzado	0% a 15%	Poca o ninguna evidencia de la consecución del atributo en el proceso valorado.
P. Parcialmente Alcanzado	>15% al 50%	Evidencia de un enfoque sistemático y de la consecución del atributo aunque algunos aspectos pueden ser impredecibles.
E. En gran Proporción Alcanzado	>51% al 85%	Evidencia de un enfoque sistemático y de una consecución significativa del atributo. La realización del proceso puede variar en algunas áreas.
C. Completamente Alcanzado	>86% al 100%	Evidencia de un enfoque completo y sistemático y de la consecución plena del atributo.

Tabla 4.5. Escala para la calificación de los atributos del proceso(s) software. Basado en [TEN06].

#### 4.3.2.6. Modelo para la Valoración de Procesos

La ISO/IEC 15504 establece que el Modelo para la Valoración de Procesos debe declarar su alcance y su cobertura en términos de:

- Los procesos a valorar tomados de un modelo de proceso de referencia
- Los niveles de capacidad seleccionados del framework de medición de referencia

La figura 4.6 muestra una perspectiva bidimensional de la capacidad del proceso. En la dimensión horizontal, denominada dimensión de proceso, se especifican las entidades de proceso que representan los procesos seleccionados del marco de proceso de referencia. La segunda dimensión se denomina dimensión de capacidad y sirve para asociar a cada entidad de proceso un nivel de capacidad de acuerdo a las consideraciones descritas en el Marco de Referencia para la Medición.

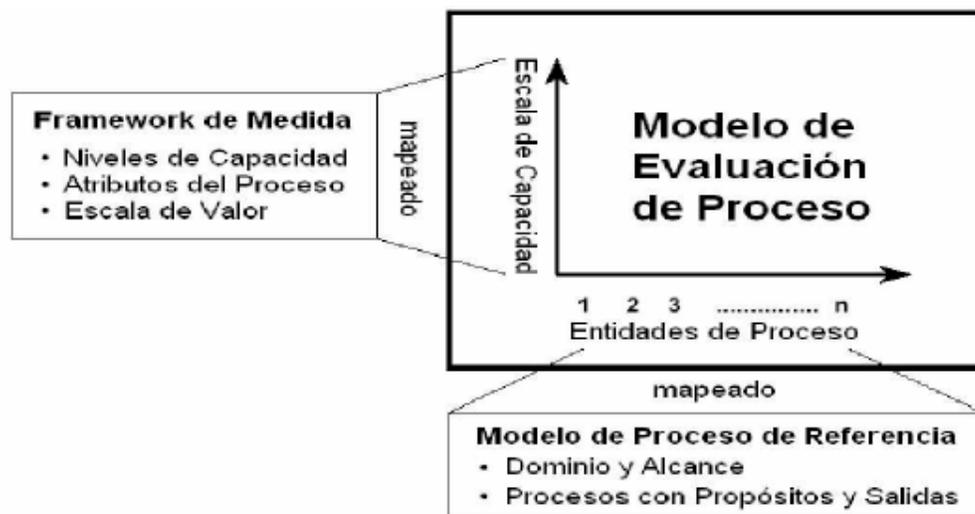


Figura 4.6. Perspectiva bidimensional del modelo del proceso de evaluación<sup>12</sup>.

<sup>12</sup> Fuente: [PIF06], p. 3.

#### **4.3.2.7. El Proceso de Valoración**

El proceso de evaluación de procesos debe ser conducido a través de una estrategia que esté en sintonía con los propósitos definidos y que se encuentre estrictamente documentada. El proceso de evaluación debe desarrollar como mínimo las siguientes actividades [ISO03]:

##### **4.3.2.7.1. Planeación**

Se debe desarrollar y documentar un plan de evaluación que incluya como mínimo:

- Las entradas requeridas especificadas en la segunda parte de la norma
- Las actividades a realizar al conducir la evaluación
- Los recursos asignados y calendario de estas actividades
- La identificación y asignación de responsabilidades de los participantes en la evaluación
- Los criterios para verificar que los requerimientos del estándar Internacional han sido cumplidos
- Una descripción de los resultados planeados de la evaluación

##### **4.3.2.7.2. Recolección de Datos**

Los datos requeridos para evaluar los procesos dentro del alcance de la evaluación y la información adicional deben ser recolectados de manera sistemática llevando a cabo como mínimo las siguientes tareas:

- La identificación de la evidencia objetiva obtenida debe ser registrada y mantenida para proporcionar la base para la verificación de las calificaciones

- La estrategia y técnicas para la selección, recolección, análisis de datos y justificación de las calificaciones deben ser explícitamente identificadas y demostrables
- Se debe establecer una correspondencia entre los procesos de las unidades administrativas de la organización, especificados en el alcance de la evaluación y los elementos del modelo del proceso de evaluación
- Cada proceso identificado dentro del alcance de la evaluación debe ser evaluado con base en evidencia objetiva
- La evidencia objetiva obtenida para cada atributo de cada proceso evaluado debe ser suficiente para lograr el propósito y alcance de la evaluación

#### 4.3.2.7.3. Validación de Datos

Los datos obtenidos deben ser validados para:

- Confirmar que la evidencia recolectada es objetiva
- Asegurar que la evidencia objetiva es suficiente y representativa para cubrir el alcance y el propósito de la evaluación
- Asegurar que los datos son completos y consistentes

#### 4.3.2.7.4. Calificación de los Atributos del Proceso

Para cada atributo del proceso se debe asignar una calificación con base en los datos validados:

- El conjunto de calificaciones de atributos del proceso se debe registrar como perfil del proceso para la unidad definida de la organización
- Durante la evaluación, se debe usar el conjunto definido de indicadores de evaluación del modelo del proceso de evaluación, el cual debe

apoyar el juicio de los evaluadores en la calificación de los atributos del proceso con el fin de proporcionar la base para la repetición entre las evaluaciones

- Se debe registrar el proceso de toma de decisión que se usará para justificar la calificación obtenida
- Se debe mantener el rastreo entre la calificación de un atributo y la evidencia objetiva que se utiliza para obtener esta calificación
- Para cada atributo del proceso calificado se debe registrar la relación entre los indicadores y la evidencia objetiva

#### 4.3.2.7.5. Generación de Reportes

Los resultados de la evaluación deben incluir como mínimo las siguientes salidas:

- La fecha de la evaluación
- Las entradas de la evaluación
- La identificación de la evidencia objetiva obtenida
- Identificación del proceso de evaluación documentado
- El conjunto de los perfiles del proceso resultantes de la evaluación (es decir, un perfil por cada proceso evaluado)
- La identificación de cualquier información adicional recolectada durante la evaluación

Adicionalmente, estas salidas deben ser documentadas y reportadas al promotor de la evaluación o a su representante asignado.

### 4.3.3. Modelos CMMI

En respuesta a la crisis percibida en torno al desarrollo de software en cuanto al incremento sustancial de los costos y los problemas de calidad, el Departamento de Defensa de los Estados Unidos estableció a principios de 1980 el Instituto de Ingeniería de Software (SEI, por sus siglas en inglés) en la Universidad Carnegie Mellon en Pittsburg, Pennsylvania. En 1988, el SEI comenzó con el desarrollo de un modelo para el mejoramiento de procesos de Ingeniería de Software, el cual fue publicado en su primera versión en Agosto de 1991 bajo el nombre de Modelo de Capacidad y Madurez para software (SW-CMM) [COC01]. A partir del desarrollo de ese modelo y hasta la fecha de hoy, han surgido otros modelos propuestos por el mismo SEI que refinan y evolucionan el trabajo original. Entre estos modelos y con mayor reconocimiento, se encuentran los modelos contenidos en el estándar CMMI.

Los modelos CMMI [SEI02] toman como base el concepto desarrollado por CMM (Capability Maturity Model) para evolucionar a un nivel que promete el crecimiento y la expansión continua de dicho concepto a múltiples disciplinas o cuerpos de conocimiento. Así como el SW-CMM [CHR03], EIA/IS 731, IPD-CMM [SEI97], SA-CMM [COF02], entre otros, los modelos CMMI son herramientas para ayudar a las organizaciones a mejorar sus procesos de desarrollo, adquisición y mantenimiento de sus productos y servicios. Además de los conceptos tradicionales de CMM como lo son la Gestión de Procesos y la Gestión de Proyectos, el modelo CMMI cubre los siguientes conceptos: Integración de Productos y Desarrollo de Procesos, Ingeniería de Sistemas, Ingeniería de Software y Relación con Proveedores.

#### **4.3.3.1. Disciplinas de CMMI**

Las disciplinas o cuerpos de conocimiento incluidos dentro de CMMI, tienen como propósito principal cubrir el desarrollo y mantenimiento de productos y servicios.

##### **4.3.3.1.1. Ingeniería de Sistemas**

La Ingeniería de Sistemas cubre el desarrollo total de sistemas, los cuales pueden o no incluir software. Los Ingenieros de Sistemas se enfocan en transformar las necesidades de los clientes, las expectativas y las restricciones en productos, y de soportar dichos productos a través de su ciclo de vida. Cuando se selecciona la Ingeniería de Sistemas como modelo, este deberá contener las áreas de proceso relativas a la Gestión de Procesos, Gestión de Proyectos, Soporte e Ingeniería.

##### **4.3.3.1.2. Ingeniería de Software**

La Ingeniería de Software cubre el desarrollo de sistemas software. Los Ingenieros de Software se enfocan en la aplicación sistemática, disciplinada y cuantificable de técnicas de desarrollo, operación y mantenimiento de software. Al igual que con la disciplina de Ingeniería de Sistemas, cuando se selecciona esta disciplina como modelo, este deberá contener las áreas de proceso relativas a la Gestión de Procesos, Gestión de Proyectos, Soporte e Ingeniería.

##### **4.3.3.1.3. Integración de Productos y Desarrollo de Procesos**

La Integración de Productos y Desarrollo de Procesos es un enfoque sistemático que tiene como propósito lograr una colaboración oportuna de los participantes claves que se ven involucrados con el producto en alguna etapa de su ciclo de vida. Lo anterior con el fin de satisfacer de la mejor manera posible las necesidades, expectativas y requisitos del cliente. Debe tenerse en

cuenta que las áreas de proceso incluídas dentro de este modelo deben estar integradas con otras áreas de proceso de otras disciplinas, ya que estas áreas de proceso por si solas, no son capaces de alcanzar los objetivos que define la disciplina. Dicho de otro modo, si una organización escoge la Integración de Productos y Desarrollo de Procesos como modelo debe, además, escoger otras(s) disciplina(s) y trabajarlas en conjunto.

#### 4.3.3.1.4. Relación con Proveedores

Cuando el esfuerzo requerido para desarrollar las actividades definidas dentro del proyecto empieza a ser sustancial, los gerentes de proyecto pueden optar por utilizar proveedores para desempeñar algunas funciones o para adicionar modificaciones a los productos que el proyecto así requiera. Cuando dichas actividades son críticas y antes que cualquier producto sea liberado, se requiere de un análisis minucioso de los proveedores y del monitoreo de las actividades que estos desempeñan. Bajo las circunstancias descritas, esta disciplina cubre la adquisición de productos provenientes de fuentes externas a la organización. Adicionalmente, y de manera similar a la disciplina de Integración de Productos y Desarrollo de Procesos, esta disciplina debe desarrollarse en conjunto con cualquier otra disciplina que se utilice para el desarrollo de productos (por ejemplo, la Ingeniería de Sistemas o la Ingeniería de Software).

#### 4.3.3.2. Estructura de CMMI

La estructura del modelo CMMI esta definida por una serie de componentes que se relacionan de manera directa con las áreas de proceso. Las áreas de proceso son el elemento más importante de la estructura de CMMI, ya que el resto de componentes que conforman dicha estructura están supeditados a ellas. A continuación se realiza una breve descripción de estos componentes:

#### 4.3.3.2.1. Áreas de Proceso

Un área de proceso es una agrupación de prácticas relacionadas en un área que, cuando son implementadas colectivamente, satisfacen un conjunto de objetivos considerados importantes para realizar mejoras significativas en dicha área [SEI02]. CMMI en su versión 1.1 define veinte y cinco áreas de proceso, las cuales son listadas a continuación:

- Administración de Acuerdos con Proveedores (SAM)
- Administración de Requisitos (REQM)
- Análisis Causal y Resolución (CAR)
- Análisis de Decisiones y Resolución (DAR)
- Aseguramiento de la Calidad de Proceso y Producto (PPQA)
- Definición del Proceso Organizacional (OPD)
- Desarrollo de Requisitos (RD)
- Desempeño del Proceso Organizacional (OPP)
- Enfoque hacia el Proceso Organizacional (OPF)
- Entorno Organizacional para Integración (OEI)
- Entrenamiento Organizacional (OT)
- Gestión de la Configuración (CM)
- Gestión de Riesgos (RSKM)
- Gestión Integrada del Proveedor (ISM)
- Gestión Integrada del Proyecto (IPM)
- Innovación Organizacional y Despliegue (OID)
- Integración de Equipos (IT)
- Integración del Producto (PI)
- Manejo Cuantitativo del Proyecto (QPM)
- Medición y Análisis (MA)
- Monitoreo y Control del Proyecto (PMC)
- Planificación del Proyecto (PP)
- Solución Técnica (TS)

- Validación (VAL)
- Verificación (VER)

#### 4.3.3.2.2. Declaraciones de Propósito

Cada área de proceso define el propósito que persigue. La declaración del propósito describe el propósito del área de proceso.

#### 4.3.3.2.3. Notas Introdutorias

Las notas de introducción del área de proceso describen los conceptos más relevantes de dicha área de proceso.

#### 4.3.3.2.4. Áreas de Proceso Relacionadas

Las áreas de proceso relacionadas son una lista que define las relaciones de alto nivel entre áreas de proceso.

#### 4.3.3.2.5. Objetivos Específicos

Un objetivo específico describe las características únicas que deben reunirse para satisfacer un área de proceso. Los objetivos específicos son utilizados en procesos de valoración para determinar si se cumple con las especificaciones de un área de proceso.

#### 4.3.3.2.6. Objetivos Genéricos

Los objetivos genéricos describen las características que deben reunirse para institucionalizar el proceso que implementa un área de proceso, y son llamados genéricos porque aplican a múltiples áreas de proceso.

#### 4.3.3.2.7. Prácticas Específicas

Una práctica específica es la descripción de una actividad que es considerada importante para poder alcanzar el objetivo específico con el cual se asocia.

#### 4.3.3.2.8. Productos de Trabajo Típicos

Los productos de trabajo típicos describen las salidas que podría tener una práctica específica, es decir, sus entregables. Estos entregables son llamados “típicos” porque puede haber otros productos de trabajo igual de efectivos pero que no se encuentran contenidos dentro del estándar.

#### 4.3.3.2.9. Subprácticas

Una subpráctica es una descripción detallada que orienta en la interpretación e implementación de una práctica específica.

#### 4.3.3.2.10. Prácticas Genéricas

Una práctica genérica es la descripción de una actividad que es considerada como importante para poder alcanzar el objetivo genérico con el cual se asocia, y son llamadas genéricas porque aparecen en múltiples áreas de proceso.

#### 4.3.3.2.11. Elaboraciones de Prácticas Genéricas

Una elaboración de práctica genérica es una descripción que brinda orientación acerca de cómo una práctica que es genérica debe ser aplicada a un área de proceso en particular.

#### 4.3.3.3. Componentes Requeridos, Esperados e Informativos

Los componentes de la estructura CMMI arriba descritos se encuentran agrupados en tres categorías que reflejan como estos componentes deben ser interpretados. La figura 4.7 muestra la relación existente entre estos elementos e indica a que categoría pertenecen.

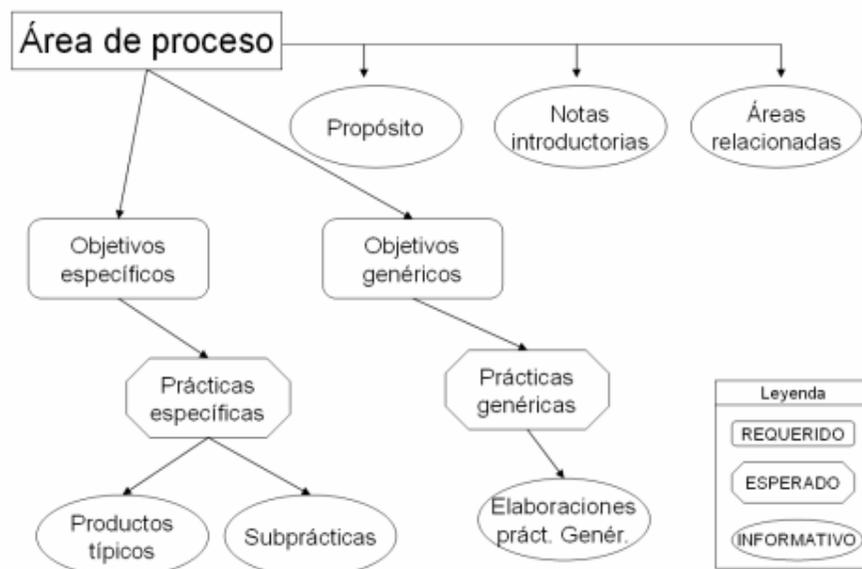


Figura 4.7. Relación entre los componentes de la estructura CMMI. Fuente [WIC06].

#### 4.3.3.3.1. Componentes Requeridos

Los componentes requeridos describen los requisitos que una organización debe reunir para satisfacer un área de proceso. La reunión de estos requisitos

debe estar visiblemente implementada en los procesos organizacionales. Los componentes requeridos en CMMI son los objetivos generales y específicos. La satisfacción de estos objetivos es utilizada en procesos de valoración para determinar si el propósito de un área de proceso ha sido alcanzado.

#### 4.3.3.3.2. Componentes Esperados

Los componentes esperados describen lo que se supone que una organización debería tener implementado para alcanzar un componente requerido. Estos componentes proveen orientación a la hora de adelantar procesos de valoración o mejoramiento de procesos. En CMMI, los componentes esperados toman la forma de prácticas específicas y prácticas genéricas, las cuales deben haber sido planeadas e implementadas en los procesos organizacionales para poder considerar que los objetivos de las áreas de proceso han sido alcanzados.

#### 4.3.3.3.3. Componentes Informativos

Los componentes informativos sirven para ayudar a las organizaciones a empezar a pensar en la forma en la que se aproximarán a los componentes requeridos y esperados. Los componentes informativos en CMMI son las subprácticas, los productos de trabajo, los amplificadores de disciplina, las elaboraciones de prácticas genéricas, los títulos y notas asociadas a las prácticas y a los objetivos, y las referencias.

#### 4.3.3.4. Representaciones

Las representaciones de CMMI pueden verse como distintas formas de articular o configurar los componentes que hacen parte del modelo. Dichas representaciones enfocan el modelo desde perspectivas diferentes, lo cual brinda a las organizaciones la oportunidad de escoger la que mejor se ajuste a

sus necesidades. En concreto, son dos las representaciones del modelo: la escalonada y la continua.

#### 4.3.3.4.1. La Representación Continua

La representación continua toma las áreas de proceso y las agrupa por categorías afines y niveles de capacidad. Los niveles de capacidad hacen referencia al cumplimiento de las prácticas genéricas y específicas de un área de proceso en particular y están diseñados para mejorar el proceso asociado a esa área de proceso. La figura 4.8 muestra la relación de los componentes del modelo bajo la representación continua.

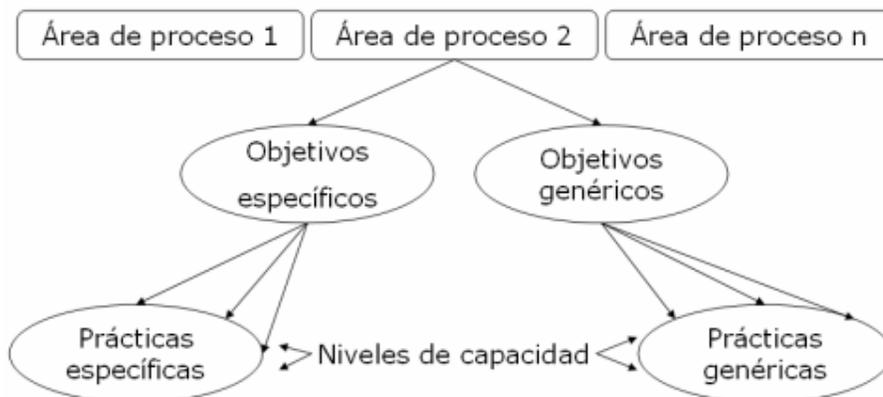


Figura 4.8. Representación continua del modelo CMMI. Fuente [WIC06].

CMMI, establece una escala jerárquica de seis niveles de capacidad, donde cero es el más bajo y representa la ejecución de un proceso que no cumple con sus objetivos y que es incapaz de producir resultados. Si un proceso alcanza los objetivos que define y la calidad de los productos que genera es óptima, puede decirse que dicho proceso ha alcanzado el nivel de capacidad más alto que, según la escala mostrada en la tabla 4.6, esta representado por el cinco.

<b>Nivel de Capacidad</b>	<b>Descripción</b>
0	Incompleto
1	Realizado
2	Gestionado
3	Definido
4	Gestionado Cuantitativamente
5	Optimizado

Tabla 4.6. Niveles de capacidad de la representación continúa.

A continuación se hace una descripción de cada uno de los niveles de capacidad [SEI02]:

Nivel 0: Incompleto. Un proceso incompleto es un proceso no realizado o realizado parcialmente. Uno o más de los objetivos específicos del área del proceso no han sido satisfechos y no existen objetivos genéricos en este nivel dado que no hay razones para institucionalizar un proceso parcialmente realizado.

Nivel 1: Realizado. Un proceso realizado es un proceso que satisface los objetivos específicos del área del proceso. El proceso soporta y habilita las actividades requeridas para generar productos de trabajo.

Nivel 2: Gestionado. Un proceso gestionado es un proceso realizado que cuenta con la infraestructura básica para soportar dicho proceso. El proceso es planeado y ejecutado de acuerdo a políticas; los empleados cuentan con las habilidades y recursos necesarios para producir salidas controladas; se involucra a los participantes clave en el proceso; el proceso es monitoreado, controlado y revisado; el proceso es evaluado por su adherencia a la descripción de proceso.

Nivel 3: Definido. Un proceso definido es un proceso Administrado que es configurado a partir del conjunto de procesos estándares de la organización de acuerdo a los parámetros e indicaciones establecidas por la misma, la cual también aporta a lo activos de proceso elementos tales como productos de trabajo, indicadores, y otra información relevante de mejoramiento de procesos.

Nivel 4: Gestionado Cuantitativamente. Un proceso Gestionado Cuantitativamente es un proceso definido y controlado usando técnicas de estadísticas y otras técnicas cuantitativas. Se establecen objetivos cuantitativos de calidad y desempeño del proceso, los cuales son usados como criterios en la administración del proceso. La calidad y el desempeño del proceso son entendidas en términos estadísticos y es manejada a través del ciclo de vida del mismo.

Nivel 5: Optimizable. Un proceso optimizable es un proceso gestionado cuantitativamente que es mejorado de acuerdo al entendimiento de las causas comunes de variación inherentes al proceso. El propósito de un proceso optimizable es el mejoramiento continuo del desempeño del proceso a través de mejoras incrementales y novedosas.

#### 4.3.3.4.2. La Representación Escalonada

La representación escalonada define niveles de madurez para clasificar a las organizaciones y preestablece una ruta a seguir para que estas adelanten paso a paso las labores de mejoramiento de sus procesos. Las áreas de proceso son agrupadas y preestablecidas en cada uno de los niveles de madurez, lo cual conforma “paquetes” o “escalones” de mejoramiento para las organizaciones. El paso por un nivel de madurez prepara a la organización a subir al siguiente nivel donde nuevas áreas de proceso deberán ser trabajadas.

A nivel estructural, mientras que en la representación continua los objetivos genéricos supeditan a las prácticas genéricas, en la representación escalonada los objetivos genéricos están relacionados con cuatro características comunes: compromiso de realización, capacidad de realización, dirigir la implementación y verificar la implementación. La figura 4.9 ilustra la interacción entre los componentes del modelo bajo esta representación.

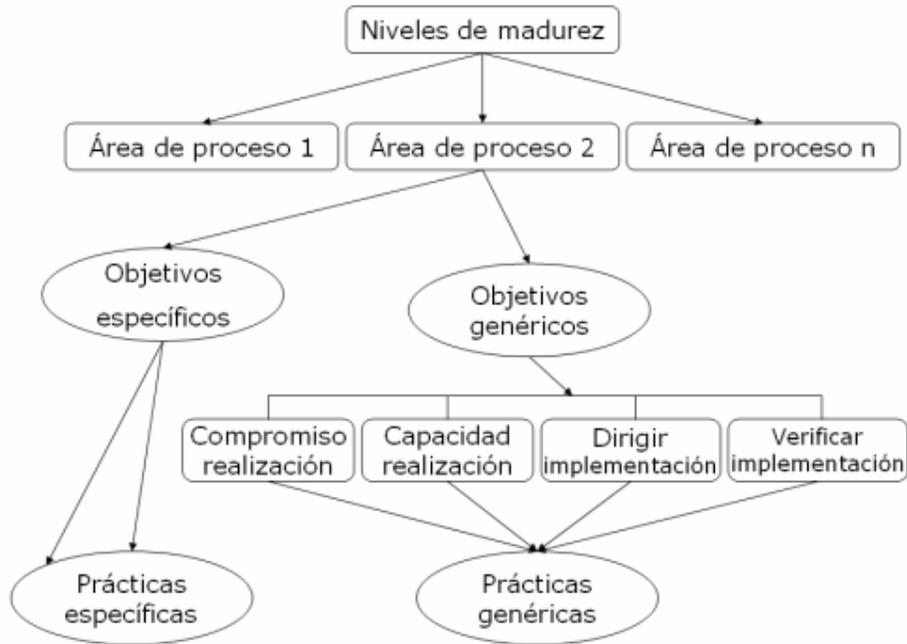


Figura 4.9. Representación escalonada del modelo CMMI. [WIC06].

CMMI, establece para la representación escalonada cinco niveles de madurez, donde el quinto nivel representa a las organizaciones con mayor madurez en el conjunto completo de sus procesos, y el nivel 1 representa a las organizaciones netamente reactivas cuyos procesos son completamente caóticos. La tabla 4.7 describe la escala de los niveles de madurez de la representación escalonada y la tabla 4.8 muestra el conjunto de áreas de procesos que pertenecen a cada nivel:

Nivel de Madurez	Descripción
1	Inicial
2	Gestionado
3	Definido
4	Gestionado Cuantitativamente
5	Optimizado

Tabla 4.7. Niveles de madurez de la representación escalonada.

<b>Nivel de Madurez</b>	<b>Área de Proceso</b>	<b>Disciplina</b>
<b>5</b>	Análisis Causal y Resolución (CAR)	Soporte
	Innovación Organizacional y Despliegue (OID)	Gestión de Procesos
<b>4</b>	Desempeño del Proceso Organizacional (OPP)	Gestión de Procesos
	Manejo Cuantitativo del Proyecto (QPM)	Gestión de Proyectos
<b>3</b>	Análisis de Decisiones y Resolución (DAR)	Soporte
	Definición del Proceso Organizacional (OPD)	Gestión de Procesos
	Desarrollo de Requisitos (RD)	Ingeniería
	Enfoque hacia el Proceso Organizacional (OPF)	Gestión de Procesos
	Entorno Organizacional para Integración (OEI)	Soporte
	Entrenamiento Organizacional (OT)	Gestión de Procesos
	Gestión de Riesgos (RSKM)	Gestión de Proyectos
	Gestión Integrada del Proveedor (ISM)	Gestión de Proyectos
	Gestión Integrada del Proyecto (IPM)	Gestión de Proyectos
	Integración de Equipos (IT)	Gestión de Proyectos
	Integración del Producto (PI)	Ingeniería
	Solución Técnica (TS)	Ingeniería
	Validación (VAL)	Ingeniería
Verificación (VER)	Ingeniería	
<b>2</b>	Administración de Acuerdos con Proveedores (SAM)	Gestión de Proyectos
	Administración de Requisitos (REQM)	Ingeniería
	Aseguramiento de la Calidad de Proceso y Producto (PPQA)	Soporte
	Gestión de la Configuración (CM)	Soporte
	Medición y Análisis (MA)	Soporte
	Monitoreo y Control del Proyecto (PMC)	Gestión de Proyectos
	Planificación del Proyecto (PP)	Gestión de Proyectos

Tabla 4.8. Conjunto de áreas de proceso por niveles de madurez.

Dado que la descripción de los niveles de madurez del 2 hasta el 5 es la misma que la descripción de los niveles de capacidad dentro de ese mismo rango, solo se describirá el primer nivel de madurez [CKS03]:

Nivel 1: Inicial. Los procesos son usualmente ad-hoc y caóticos. La organización usualmente no provee un ambiente estable para soportar los procesos. El éxito en la organización depende de las competencias y actos heroicos de la gente y no del uso de procesos probados. En medio de este caos, las empresas producen productos y servicios que trabajan, sin embargo, la producción se excede en sus costos y no cumple con los cronogramas. Una empresa en nivel 1 se caracteriza por la tendencia a sobrecargarse, abandonar procesos en tiempo de crisis y con muy baja capacidad de repetir sus éxitos.

#### 4.3.3.5. Comparativo entre ambas Representaciones

Existen muchos criterios válidos para escoger o descartar una representación. Según [CKS03], si no se tiene experiencia con ninguna de la dos representaciones es indiferente la decisión que se tome ya que el riesgo de equivocación simplemente no existe. En contraposición, si alguna vez se ha utilizado CMM y si se es familiar con una representación en particular, la sugerencia es seguir utilizando esa representación. De todas maneras, la escogencia de una u otra representación depende de las características particulares de cada proyecto, de las necesidades de negocio y de la cultura organizacional que se tenga. La tabla 4.9 resume algunos criterios que pueden ayudar a filtrar esta decisión:

<b>Representación Continua</b>	<b>Representación Escalonada</b>
Da la libertad de seleccionar el orden de mejora que mejor reúna los objetivos de negocio de la organización y mitiga las áreas de riesgo de la misma.	Proporciona a las organizaciones un camino de mejora predefinido y probado.
Da visibilidad del nivel de capacidad de proceso alcanzado en cada área de proceso.	Se enfoca sobre un conjunto de procesos que proveen una organización con una capacidad específica que esta caracterizada por cada nivel de madurez.
Provee una técnica de calificación del nivel de capacidad que es usada para mejoramiento dentro de la organización y que rara vez es comunicada externamente.	Provee una técnica de calificación del nivel de madurez que es frecuentemente usada en la comunicación interna y externa de la organización, y durante adquisiciones como técnica para calificar proveedores.
Permite mejoras de diferentes procesos a diferentes escalas (nivel de capacidad que se quiere alcanzar).	Sintetiza los resultados de un proceso de mejoramiento limitándolos a un solo número que representa un nivel de madurez.
Es un nuevo enfoque que aún no tiene datos suficientes que demuestren el retorno de la inversión (ROI).	Es un enfoque respaldado por un historial de uso relativamente largo, lo que incluye casos de estudio e información que demuestra el retorno de la inversión.
Permite una fácil migración de SECM (EIA/IS 731).	Provee una fácil migración de CMM a CMMI.

Tabla 4.9. Comparativo entre las representaciones continua y escalonada. Basado en [UNC05].

<b>Representación Continua</b>	<b>Representación Escalonada</b>
Proporciona una fácil comparación con el modelo de mejoramiento ISO/IEC 15504 debido a que la organización de las áreas del proceso es derivada de este modelo.	Permite la comparación con la ISO/IEC 15504, pero la organización de las áreas del proceso no corresponde con la organización utilizada en esa norma.

Tabla 4.9. Comparativo entre las representaciones continua y escalonada. Basado en [UNC05].

#### **4.3.3.6. Áreas de Proceso CMMI**

Las áreas de proceso CMMI pueden ser agrupadas en las siguientes categorías:

- Gestión de Procesos
- Gestión de Proyectos
- Ingeniería
- Soporte

Las tablas 4.10, 4.11, 4.12 y 4.13, describen el propósito y los elementos clave de cada una de las áreas de proceso que pertenecen a las categorías arriba mencionadas.

#### 4.4. Gestión de Procesos

Son las áreas de proceso que tienen que ver con la definición, planeación, despliegue, implementación, monitoreo, control, medición y mejoramiento de procesos. La tabla 4.10 describe cuales son estas áreas.

Área de Proceso	Propósito	Descripción
<i><u>Enfoque hacia el Proceso Organizacional (OPF)</u></i>	Planear e implementar el mejoramiento del proceso organizacional basado en las fortalezas y debilidades actuales de los activos de proceso y los procesos de la organización.	El mejoramiento ocurre en el contexto de las necesidades de la organización y es usado para orientarlas hacia sus objetivos. La organización motiva la participación en procesos de mejoramiento a quienes desempeñaran los procesos a mejorar. De este modo, el grupo alentado se encarga de coordinar y liderar las actividades de evaluación y de mejora. Esto involucra desarrollar un plan de mejora (p.e. IDEAL), el cual debe incluir planes de evaluación, de acciones, de pilotos y de despliegue.
<i><u>Definición del Proceso Organizacional (OPD)</u></i>	Establecer y mantener un conjunto de activos de procesos organizacionales usables.	Los activos de proceso de la organización posibilitan una ejecución consistente del proceso a través de la organización y proveen una base acumulativa a la organización con beneficios a largo plazo. La librería de activos de proceso de la organización es una colección de ítems mantenidos por la organización para ser usados por los proyectos y las personas dentro de la organización. Esta colección de ítems incluye descripciones de procesos, elementos de procesos, descripciones de modelos de ciclo de vida, guías de instanciación, documentación relacionada con el proyecto e información. La librería de activos de procesos soporta el aprendizaje organizacional permitiendo compartir las mejores prácticas a través de la organización. Un proceso estándar está compuesto de otros procesos o elementos de procesos (unidad fundamental de definición de proceso, descrita por actividades y tareas para realizar un trabajo). La arquitectura del proceso provee las reglas para la conexión de los elementos del proceso de un proceso estándar.

Tabla 4.10. Áreas de proceso relacionadas con la Gestión de Procesos. Adaptado de [UNC05].

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<u>Entrenamiento Organizacional (OT)</u>	Desarrollar las habilidades y el conocimiento de las personas para que puedan desempeñar sus roles efectiva y eficientemente.	Incluye el entrenamiento necesario para soportar los objetivos estratégicos de la organización y las necesidades tácticas que son comunes a los grupos de proyectos y de soporte. Las necesidades individuales y específicas de entrenamiento por cada proyecto y grupos de soporte se obtienen del alcance de ésta área. Un programa de entrenamiento organizacional involucra: identificar las necesidades de entrenamiento de la organización, obtener y proveer entrenamiento dirigido a estas necesidades, establecer y mantener la capacidad de entrenamiento, establecer y mantener los registros de entrenamiento, y asegurar la efectividad del entrenamiento.
<u>Desempeño del Proceso Organizacional (OPP)</u>	Establecer y mantener una comprensión cuantitativa del desempeño del conjunto de activos de proceso estándar de la organización para soportar objetivos de calidad y desempeño de los procesos, y para proveerla de información del desempeño del proceso, líneas base y modelos cuantitativamente gestionados.	El desempeño del proceso es una medida de los resultados reales logrados al seguir un proceso. Este desempeño es caracterizado tanto por medidas del proceso (esfuerzo, tiempo, etc.), como por medidas del producto (confiabilidad, densidad de defectos, etc.). Esta información es analizada para establecer la distribución y el rango de resultados, los cuales indicarán el rango esperado de desempeño cuando sea utilizado por un proyecto en la organización.
<u>Innovación Organizacional y Despliegue (OID)</u>	Seleccionar y desplegar mejoras incrementales e innovadoras que mejoren de manera medible el proceso organizacional y las tecnologías. El mejoramiento soporta los objetivos de calidad y desempeño de procesos de la organización que son derivados de los objetivos de negocio de la misma.	Permite la selección e implementación de mejoras a la organización que permiten reunir las capacidades necesarias para alcanzar sus objetivos de calidad y de desempeño del proceso. Las mejoras en esta área se refieren a ideas probadas que pueden cambiar los procesos y tecnologías de la organización para cumplir de una mejor forma con sus objetivos de calidad y desempeño. Esos objetivos son: mejorar la calidad del producto, incrementar la productividad, disminuir el tiempo de los ciclos, alcanzar una gran satisfacción del cliente y del usuario final, acortar tiempos de desarrollo o de producción para cambiar funcionalidad, adicionar características o adaptarse a nuevas tecnologías.

Tabla 4.10. Áreas de proceso relacionadas con la Gestión de Procesos. Adaptado de [UNC05].

#### 4.5. Gestión de Proyectos

Son las áreas de proceso que tienen que ver con la planeación, monitoreo y control de proyectos.

Área de Proceso	Propósito	Descripción
<u>Planificación del Proyecto (PP)</u>	Establecer y mantener planes que definan las actividades del proyecto.	La planificación empieza con los requisitos del proyecto e incluye estimar atributos del producto y de las tareas, determinar los recursos necesarios, negociar los compromisos, realizar cronogramas e identificar y analizar los riesgos del proyecto. El plan provee la base para ejecutar y controlar las actividades del proyecto para cumplir con el compromiso establecido con el cliente. Esto involucra desarrollar un plan, teniendo en cuenta la interacción con todos los participantes, obtener los compromisos del plan y mantenerlo a lo largo del proyecto.
<u>Monitoreo y Control del Proyecto (PMC)</u>	Proveer la información acerca del progreso del proyecto para poder tomar las acciones correctivas apropiadas cuando la ejecución del proyecto se desvía significativamente respecto del plan.	La base del seguimiento y control, es un plan de proyecto documentado, a partir del cual se comunica el estado y se toman acciones correctivas. El progreso se determina básicamente comparando el trabajo real a partir de los atributos de productos y tareas con respecto al trabajo ideal definido en el cronograma o el diagrama de seguimiento seleccionado. Una desviación significativa (si no es resuelta amenaza los objetivos del proyecto) deberá ser detectada y corregida oportunamente. Normalmente se requiere revisar los planes y adecuarlos al desarrollo del proyecto.
<u>Administración de Acuerdos con Proveedores (SAM)</u>	Gestionar la adquisición de productos desde proveedores con los cuales existen acuerdos formales.	Esta área aplica principalmente a la adquisición de productos y componentes que son liberados al cliente del proyecto. Para minimizar los riesgos del proyecto, esta área también debe aplicarse cuando los productos no son liberados al cliente (procesos, ambientes de desarrollo, herramientas de pruebas, etc.). Esta área no aplica cuando el proveedor hace parte del proyecto.

Tabla 4.11. Áreas de proceso relacionadas con la Gestión de Proyectos. Adaptado de [UNC05].

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<u><i>Gestión Integrada del Proyecto (IPM)</i></u>	Establecer y mantener el proyecto y el contexto de los participantes relevantes de acuerdo a un proceso definido e integrado que es configurado desde un conjunto de procesos estándares de la organización.	Involucra establecer el proceso definido del proyecto a partir de los procesos estándares de la empresa, gestionar el proyecto usando el proceso definido para el proyecto, usar y contribuir a los activos de proceso organizacionales, permitir a los intereses relevantes de cada participante: ser identificados, considerados y si es apropiado, orientarlos durante el desarrollo del producto (vistas por roles). También involucra asegurar que los participantes realizan sus tareas de manera coordinada y oportuna para: (1) lograr las metas del producto y del proyecto, (2) Cumplir los compromisos e (3) Identificar, seguir y resolver inconvenientes.
<u><i>Gestión de Riesgos (RSKM)</i></u>	Identificar problemas potenciales antes que estos ocurran con el fin de planear y realizar actividades de manejo de riesgo.	Se aplica un enfoque continuo de gestión del riesgo para anticipar y mitigar efectivamente los riesgos que puedan tener un impacto crítico sobre el proyecto. Esta gestión incluye una identificación temprana y agresiva del riesgo con la participación de todos los participantes involucrados, para lo cual el líder deberá proveer un ambiente abierto y libre para su discusión. La gestión del riesgo puede ser dividida en tres partes: definir la estrategia; identificar y analizar riesgos, y manejar los riesgos identificados, incluyendo los planes de mitigación del riesgo cuando estos lo ameriten.
<u><i>Integración de Equipos (IT)</i></u>	Formar y mantener un grupo integrado para el desarrollo de los productos de trabajo.	Un equipo integrado está compuesto de participantes quienes generan e implementan decisiones para el producto de trabajo en desarrollo. Los miembros del equipo son colectivamente responsables de la liberación del producto de trabajo. Esta área busca que los miembros tengan las habilidades y experticia necesarias, para que puedan seguir todo el ciclo de vida del producto, se colaboren interna y externamente, se entiendan con unos objetivos y tareas comunes, se conduzcan a sí mismos (independencia) de acuerdo a unos principios operativos y reglas fundamentales.

Tabla 4.11. Áreas de proceso relacionadas con la Gestión de Proyectos. Adaptado de [UNC05].

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<i><u>Gestión Integrada del Proveedor (ISM)</u></i>	Identificar proactivamente fuentes de productos que puedan ser usadas para satisfacer los requisitos del proyecto y gestionar los proveedores seleccionados mientras se mantiene una relación cooperativa proyecto-proveedor.	Esta área se construye sobre el área de proceso SAM, pero adicionando las prácticas que enfatizan en la relación cooperativa entre los proveedores. Está diseñada para las situaciones en las cuales los proyectos usan proveedores para realizar las funciones que son críticas para el éxito del proyecto. Esta área involucra actividades tales como: identificar, analizar y seleccionar fuentes potenciales de productos; evaluar y determinar las fuentes usadas para la adquisición de productos, monitorear y analizar los proveedores seleccionados, evaluar estos productos y revisar los contratos o las relaciones con los proveedores cuando sea necesario.
<i><u>Manejo Cuantitativo del Proyecto (QPM)</u></i>	Gestionar cuantitativamente el proceso definido para el proyecto para lograr los objetivos establecidos de calidad y desempeño del mismo.	Involucra establecer estos objetivos, identificar los subprocesos que componen el proceso del proyecto basado en la estabilidad y capacidad histórica de la información encontrada en las líneas base o modelos de desempeño del proceso. Se deben seleccionar estos subprocesos de manera estadística. Se hace seguimiento y control cuantitativo, para lo cual se selecciona técnicas de medida y análisis para la gestión de los subprocesos. Se definen las variaciones de estos procesos, se monitorea el desempeño de los procesos en la satisfacción de los objetivos cuantitativos. Se registra la información de gestión de calidad y las estadísticas en el repositorio que disponga la organización.

Tabla 4.11. Áreas de proceso relacionadas con la Gestión de Proyectos. Adaptado de [UNC05].

#### 4.6. Ingeniería

Las áreas de proceso de Ingeniería cubren las actividades de desarrollo y mantenimiento de cualquier producto o servicio que se encuentren dentro del dominio de esa disciplina.

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<i><u>Desarrollo de Requisitos (RD)</u></i>	Producir y analizar los requisitos del cliente, del producto y de sus componentes.	Los requisitos deben ser elicitados, analizados, validados a partir de las necesidades, expectativas y restricciones del cliente. Se debe recolectar y coordinar las necesidades de los participantes y desarrollar el ciclo de vida de los requisitos del producto. Se deben establecer los requisitos iniciales del producto y de sus componentes de manera consistente con los requisitos del cliente. Esta área se orienta a los requisitos del cliente antes que a los requisitos de producto. Los requisitos del producto son refinados a partir de los requisitos del cliente y derivados de la solución de diseño seleccionada. Los requisitos son identificados, definidos y actualizados a través de las fases del ciclo de vida del producto.
<i><u>Administración de Requisitos (REQM)</u></i>	Administrar los requisitos de los productos y componentes de producto e identificar inconsistencias entre los requisitos y los planes del proyecto y los productos de trabajo.	Esta área es la encargada de gestionar todos los requisitos recibidos o generados por el proyecto, incluyendo requisitos técnicos y no técnicos. Cuando el área de desarrollo genera requisitos de producto o de componentes del producto, estos también deberán ser administrados por esta área. El proyecto sigue los pasos apropiados para asegurar que hay una adecuada gestión de requisitos para soportar las necesidades de planeación y ejecución del proyecto. Cuando un proyecto recibe los requisitos desde un proveedor de requisitos, los requisitos son revisados con el proveedor para resolver problemas y malos entendidos antes de ser incorporados a los planes del proyecto. Una vez son incorporados, el proyecto deberá gestionar el cambio en estos cuando se identifiquen inconsistencias entre planes, productos de trabajo y requisitos. Parte de la gestión de requisitos es documentar sus cambios y sus razones, y mantener la trazabilidad bidireccional entre fuentes de requisitos y todos los requisitos del producto y componentes de producto.

Tabla 4.12. Áreas de proceso relacionadas con la Ingeniería. Adaptado de [UNC05].

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<u>Solución Técnica (TS)</u>	Diseñar, desarrollar e implementar soluciones de acuerdo a los requisitos. Las soluciones, diseños e implementaciones se refieren a componentes de producto, productos, procesos relacionados con el ciclo de vida del producto o combinaciones apropiadas de estos.	Es un área aplicable a cualquier nivel: arquitectura de producto, componentes de producto, producto, proceso relacionado al ciclo de vida del producto y servicios. Esto incluye evaluar y seleccionar soluciones que satisfagan potencialmente los requisitos definidos, desarrollar diseños detallados de las soluciones seleccionadas e implementar los diseños como productos o componentes de producto. Normalmente estas actividades se soportan interactivamente unas a otras. Se pueden utilizar prototipos para ir adquiriendo conocimiento de la solución técnica o de un paquete completo de requisitos.
<u>Integración del Producto (PI)</u>	Ensamblar el producto desde los componentes del producto, asegurar que el producto integrado funciona adecuadamente para ser entregable.	Está orientada a la integración de componentes en componentes más complejos o en un producto completo. El alcance de esta área es lograr la integración del producto a través de un ensamble progresivo e iterativo de componentes, de una manera incremental de acuerdo a una secuencia y unos procedimientos de integración definidos. Un aspecto crítico en el área es la gestión de las interfaces internas y externas de los productos y sus componentes para asegurar su compatibilidad.
<u>Verificación (VER)</u>	Asegurar que los productos de trabajo seleccionados cumplen con los requisitos especificados.	Incluye la preparación y ejecución de la verificación, así como la identificación de acciones correctivas. Esta se aplica tanto al producto como a los productos de trabajo intermedios de acuerdo a los requisitos del cliente, del producto y de los componentes del producto. La verificación es un proceso incremental debido a que ocurre a través de todo el proceso de desarrollo, pues se aplica a los requisitos, productos de trabajo, componentes de productos y productos.

Tabla 4.12. Áreas de proceso relacionadas con la Ingeniería. Adaptado de [UNC05].

Área de Proceso	Propósito	Descripción
<u>Validación (VAL)</u>	Demostrar que el producto o componentes del producto cumplen completamente con su propósito de uso cuando es instalado en su ambiente de trabajo.	Es aplicable a todos los aspectos del producto en cualquiera de sus ambientes de trabajo, tales como operación, entrenamiento, manufactura, mantenimiento y servicio de soporte. Los métodos de validación deben ser aplicados tanto a los productos de trabajo como a los productos y a los componentes de productos. El ambiente de validación debe representar el ambiente de trabajo sobre el cual se quiere probar. Mientras la verificación se orienta a que los productos de trabajo reflejen los requisitos especificados, la validación demuestra que este cumple completamente con su propósito de uso. Las actividades de validación son similares a las de verificación, pero en estas, frecuentemente son involucrados los usuarios finales. Puede ejecutarse en conjunto con la verificación y usar el mismo ambiente de pruebas.

Tabla 4.12. Áreas de proceso relacionadas con la Ingeniería. Adaptado de [UNC05].

#### 4.7. Soporte

Las áreas de proceso de soporte cubren las actividades que apoyan el desarrollo y mantenimiento de productos, y están dirigidas a complementar, mejorar o ayudar a procesos que aplican de manera más general a la organización.

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<u><i>Gestión de la Configuración (CM)</i></u>	Establecer y mantener la integridad de los productos de trabajo usando identificación de la configuración, control de la configuración, estado de cuenta de la configuración, y las auditorías de configuración.	El área involucra identificar la configuración de los productos de trabajo seleccionados para realizar la línea base en ciertos momentos. Controlar los cambios de los ítems de configuración, construir o proveer especificaciones para productos de trabajo desde un sistema de gestión de la configuración. Mantener la integridad de las líneas base y proveer el estado de cuenta y la información de la configuración a los desarrolladores, usuarios finales y a los clientes.
<u><i>Aseguramiento de la Calidad de Proceso y Producto (PPQA)</i></u>	Proveer un grupo de trabajo con la suficiente autonomía para revisar procesos internos y productos de trabajo.	Esta área involucra evaluar objetivamente los procesos, productos de trabajo y servicios. Identificar y documentar no conformidades. Proveer retroalimentación al grupo del proyecto y a los directores sobre los resultados de las actividades de calidad y asegurar que las no conformidades sean resueltas. Mientras el área de Verificación asegura que los requisitos especificados son satisfechos, ésta área asegura que los procesos planeados son realizados.
<u><i>Medición y Análisis (MA)</i></u>	Desarrollar y sostener una capacidad de medición que se utiliza para soportar las necesidades de información de la gestión de procesos.	Se debe especificar los objetivos de la medición y el análisis, de acuerdo a los objetivos y necesidades de información identificados. Se debe especificar las medidas, recolección de datos, mecanismos de almacenamiento, técnicas de análisis, y mecanismos de reporte y de retroalimentación. Implementar la recolección, almacenamiento, análisis y reporte de la información. Proveer resultados objetivos que pueden ser usados para tomar las acciones correctivas apropiadas.

Tabla 4.13. Áreas de proceso relacionadas con el Soporte. Adaptado de [UNC05].

<b>Área de Proceso</b>	<b>Propósito</b>	<b>Descripción</b>
<i><u>Entorno Organizacional para Integración (OEI)</u></i>	Proveer una infraestructura y personal de gestión para la integración del producto y el desarrollo de procesos (IPPD).	La organización es un sistema integrado capaz de proveer y sostener las personas, los productos y procesos necesarios para una ejecución efectiva y eficiente de sus proyectos. La organización debe proveer los estímulos necesarios para promover la excelencia personal y de los grupos. Una característica importante de ambientes efectivos para la integración incluye gente entrenada para explotar el ambiente colaborativo; un lugar de trabajo que provea los recursos para maximizar la productividad de la gente y facilitar los grupos integrados; y un conjunto de activos de procesos estándar y organizacionales que culturalmente dispongan de un ambiente IPPD que promueva y premie la excelencia de grupos, así como la individual.
<i><u>Análisis de Decisiones y Resolución (DAR)</u></i>	Analizar posibles decisiones utilizando un proceso de evaluación formal que evalúa alternativas identificadas a partir de unos criterios establecidos.	Involucra establecer guías para determinar que temas o problemas requieren de un proceso de evaluación formal. Un proceso de evaluación formal es un enfoque estructurado de evaluar alternativas de solución de acuerdo a unos criterios establecidos, para determinar una solución recomendada. Esta área incluye establecer los criterios, identificar las alternativas de solución, seleccionar los métodos de evaluación, evaluar las alternativas usando los criterios y métodos establecidos y seleccionar las soluciones recomendadas de acuerdo a los resultados de la evaluación.
<i><u>Análisis Causal y Resolución (CAR)</u></i>	Identificar las causas de defectos y otros problemas y tomar la acción de prevenir su ocurrencia en el futuro.	Esta área de proceso involucra identificar y analizar las causas de defectos y otros problemas, así como tomar las acciones específicas para eliminar las causas y prevenir la ocurrencia de este tipo de defectos o de problemas en el futuro. El análisis y la resolución causal mejora la calidad y la productividad previniendo la introducción de defectos en el producto. Es un mecanismo de comunicar lecciones aprendidas entre los proyectos. Esta área nos solo aplica a defectos sino a otros problemas, por ejemplo puede ser utilizado para mejorar atributos de calidad (desempeño).

Tabla 4.13. Áreas de proceso relacionadas con el Soporte. Adaptado de [UNC05].

#### 4.7.1. Modelo CMMI-AM

El Módulo de Adquisición de CMMI (CMMI-AM, por sus siglas en inglés) fue desarrollado por el SEI como una extensión de CMMI y su liberación se produjo en Mayo de 2005. Básicamente, CMMI-AM amplía a CMMI identificando las prácticas relativas al tema de la adquisición de productos y servicios sin dejar de lado, claro esta, aquellas que de cara a clientes y usuarios conciernen y caracterizan el tema del desarrollo de software.

Del mismo modo que CMMI toma como base los modelos CMM, CMMI-AM se fundamenta, resume y explota los siguientes modelos:

- Modelo de Capacidad y Madurez para la Adquisición de Software (SA-CMM, por sus siglas en inglés) [COF02]
- CMMI
- Modelo de Capacidad y Madurez Integrado de la FAA (FAA-iCMM, por sus siglas en inglés)<sup>13</sup>

Respecto del CMMI “puro”, CMMI-AM fue desarrollado para ser utilizado de manera conjunta con dicho estándar y puede ser visto como un modelo que ajusta y que permite entender a CMMI en el contexto de los entornos de adquisición<sup>14</sup>.

---

<sup>13</sup> Esta es una variación del modelo CMM introducida por la Administración Federal de Aviación de los Estados Unidos (FAA, por sus siglas en inglés).

<sup>14</sup> [GAL06], p. 31.

#### 4.7.2. Estructura del Modelo

Es importante tener en cuenta que CMMI-AM se focaliza en aquellas actividades y mejores prácticas que son desarrolladas a primer nivel dentro del proyecto, es decir, se centraliza en los procesos que son de cara a clientes, proveedores y usuarios, dentro de los que se incluyen aquellos que tendrán la responsabilidad de soportar y mantener el producto. En este sentido, el modelo extrae de CMMI una serie de áreas de proceso de las cuales, en muchos casos, se puede discernir entre las responsabilidades de la organización contratante y las del proveedor.

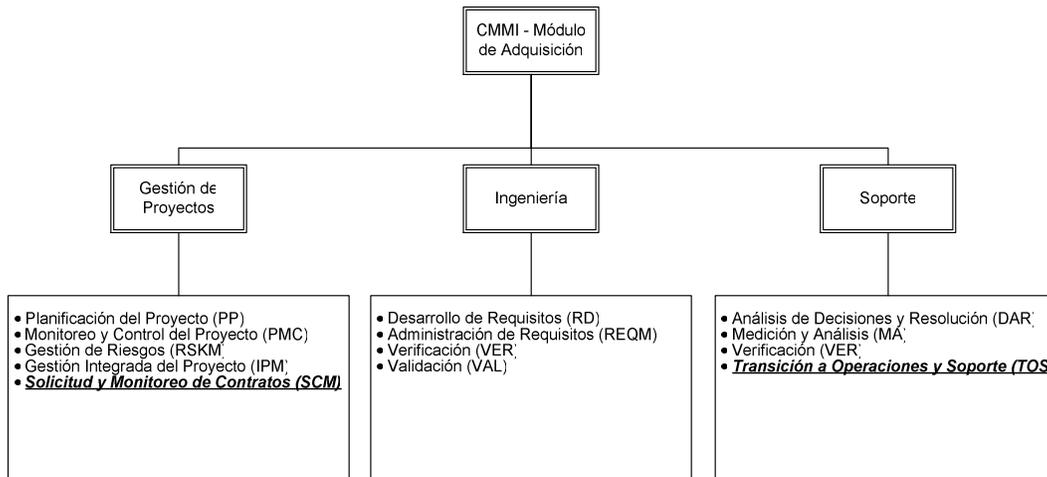


Figura 4.10. Clasificación de las áreas de proceso de CMMI-AM.

Como se puede ver en la figura 4.10, CMMI-AM identifica doce áreas de proceso. Diez de estas áreas están definidas en CMMI, mientras que las otras dos son extensiones del estándar para el caso de la adquisición. Adicionalmente, y de la misma forma en que lo hace CMMI, estas áreas de proceso son clasificadas de acuerdo a su naturaleza y entran a conformar los grupos de Gestión de Proyectos, Ingeniería y Soporte. Debe notarse que CMMI-AM no incluye áreas de proceso relativas a la Gestión de Procesos. Las áreas de proceso adicionadas son:

- Solicitud y Monitoreo de Contratos
- Transición a Operaciones y Soporte

A continuación se describen brevemente estas nuevas áreas de proceso [SEI05]:

#### **4.7.2.1. Solicitud y Monitoreo de Contratos**

El propósito de esta área de proceso es preparar una solicitud de propuesta que identifique las necesidades de una adquisición en particular, para seleccionar el proveedor que mejor sea capaz de satisfacer esas necesidades, y establecer el proceso para monitorear el proveedor durante la duración del contrato.

Los objetivos y prácticas específicas relacionadas con esta área de proceso, son listados a continuación:

*El proyecto esta preparado para conducir la solicitud*

- Designar un oficial en jefe responsable de tomar la decisión de selección.
- Establecer y mantener un paquete de solicitud que incluya las necesidades de la adquisición y los criterios de evaluación propuestos
- Establecer y mantener independientemente revisados los costos y estimados de tiempos de los productos a ser adquiridos
- Validar el paquete de solicitud con los usuarios finales y con los posibles proveedores para asegurar que los estimados realizados en cuanto a enfoque, costos y tiempos, son realistas y que de manera razonable conducen a un producto usable

*Los proveedores son seleccionados con base en el paquete de solicitud*

- Evaluar proveedores de acuerdo a los criterios de evaluación documentados
- Utilizar los resultados de la evaluación como base para soportar las decisiones de selección

*Los contratos son creados con base en las necesidades de la adquisición y en los enfoques propuestos por los posibles proveedores*

- Establecer y mantener un mutuo entendimiento con los proveedores seleccionados y los usuarios finales tomando como base las necesidades de la adquisición y los enfoques propuestos por el proveedor
- Establecer y mantener procesos y procedimientos para la comunicación con los proveedores que enfatizan en las necesidades, expectativas e indicadores de efectividad a ser usados a través de la adquisición

*El trabajo es coordinado con los proveedores para asegurar que el contrato es ejecutado apropiadamente*

- Monitorizar y evaluar los procesos seleccionados y usados por los proveedores tomando como base la documentación de procesos proveída por los mismos
- Evaluar los productos de trabajo seleccionados del proveedor tomando como base los criterios de evaluación documentados
- Revisar los acuerdos establecidos con el proveedor y la relación existente con este, como estrategia apropiada para reflejar cambios en las condiciones

#### **4.7.2.2. Transición a Operaciones y Soporte**

El propósito de la transición a operaciones y soporte es facilitar la transición del producto al usuario final y al eventual equipo que la soportará, y planear el ciclo de vida de evolución del mismo.

A continuación se listan los objetivos y prácticas específicas relacionadas con esta área de proceso:

*Se realiza la preparación para la transición a operaciones y soporte*

- Establecer y mantener una estrategia para la transición a operaciones y soporte
- Establecer y mantener planes para pasar los productos adquiridos hacia su uso operativo y soporte
- Establecer y mantener los requerimientos de capacitación para el personal de soporte
- Establecer y mantener los requerimientos iniciales y futuros de recursos para desarrollar las actividades de operación y soporte
- Identificar y asignar responsabilidad organizacional para el soporte
- Establecer y mantener criterios para asignar responsabilidades para realizar labores de mejoramiento
- Establecer y mantener criterios de transición para los productos adquiridos

*Las decisiones y acciones respecto a la transición son ejecutadas de acuerdo con los criterios establecidos*

- Evaluar el estado de preparación de los productos adquiridos para ser sometidos a la transición a operaciones y servicio

- Evaluar el estado de preparación del personal de operaciones y soporte que asumirá la responsabilidad por los productos adquiridos
- Analizar los resultados de todas las actividades de transición e identificar las acciones correctivas apropiadas

#### **4.7.2.3. Prácticas Genéricas**

Como se mencionó en el literal “Prácticas Genéricas” (ver sección 4.3.3.2.10), las prácticas genéricas son “mejores prácticas” que aplican a cualquier área de proceso, y pueden ser vistas como mecanismos que aseguran que las prácticas específicas asociadas a un área en particular son realmente llevadas a cabo. A continuación se describe la lista de prácticas genéricas definidas en CMMI y que de igual modo aplican a todas las áreas de proceso incluídas en CMMI-AM:

- Establecer y mantener una política organizacional para planear y desarrollar los procesos.
- Establecer y mantener el plan para desarrollar el proceso.
- Proveer los recursos adecuados para desempeñar el proceso, desarrollando los productos de trabajo, y ofreciendo los servicios del proceso.
- Asignar responsabilidad y autoridad para desempeñar el proceso, desarrollar los productos de trabajo, y proveer los servicios del proceso.
- Capacitar al personal que desempeñará o soportará el proceso según las necesidades.

- Poner los productos de trabajo generados por el proceso bajo los niveles apropiados de gestión de la configuración.
- Identificar e involucrar los participantes relevantes tal como se especifica en el plan.
- Monitorear y controlar el proceso a través del plan de desempeño del mismo y tomar las acciones correctivas apropiadas.
- Evaluar objetivamente la adherencia del proceso a su descripción, estándares, procedimientos, y manejar las no conformidades.
- Revisar las actividades, el estado, y los resultados del proceso con una gestión de alto nivel y resolver problemas.
- Establecer y mantener la descripción de un proceso definido.
- Recopilar productos de trabajo, mediciones, resultados de mediciones, e información de mejoramiento derivada de planear y desempeñar el proceso, con el fin de soportar su uso futuro y mejorar los procesos organizacionales y los activos de proceso.

Como se verá mas adelante, este trabajo toma muchos de los elementos contenidos en CMMI y en CMMI-AM como base para desarrollar la estrategia que se propone.

#### **4.8. El Desarrollo de Software en el Contexto de la Delegación**

En las secciones anteriores se describió teóricamente los elementos más importantes que tienen que ver con la tercerización, el desarrollo de software y la calidad en el desarrollo de software. Este análisis se realizó de manera

independiente, como si cada uno de esos temas hubieran sido aislados y respetando los límites conceptuales que cada uno define. Ahora bien, en el “mundo real” y bajo el contexto de la problemática y de la estrategia que este trabajo de grado pretende presentar, los conceptos analizados anteriormente no se encuentran aislados y, por el contrario, mantienen una fuerte relación. Es propósito de esta sección establecer la manera en que los elementos descritos se relacionan y describir las problemáticas que surgen de esas relaciones. Para propósitos de evaluación, es preciso anotar que los conceptos que aquí se describen argumentan la problemática planteada al inicio de este trabajo y le dan mas fuerza a la justificación del mismo.

#### **4.8.1. Relación entre el Cliente y el Proveedor**

Como se describió en detalle en la sección “La Tercerización”, son muchas las razones válidas para que una organización quiera delegar en un proveedor una función u operación de negocio. En el caso concreto de los proyectos de desarrollo de software, una organización, a la cual se seguirá haciendo referencia como la parte cliente, podría exponer razones tales como:

- Es difícil conseguir y contratar buenos desarrolladores
- Es más barato tercerizar que contratar buenos desarrolladores
- Alguien mas puede desarrollar el producto requerido de manera mas rápida, económica y con una mayor calidad
- Se carece de las habilidades y de los recursos requeridos para desarrollar el producto
- Se desea disponer del recurso interno para labores que se consideran mas importantes

En contraposición a lo anterior, delegando un proyecto de desarrollo de software se podrían presentar inconvenientes como:

- Pérdida de control sobre el proceso de desarrollo de software lo cual aumenta de manera considerable el riesgo del proyecto; posiblemente se coloque en las manos del proveedor el destino de la competitividad de la organización<sup>15</sup>
- No se desarrollan las competencias internas para desarrollar software; estas pueden requerirse en situaciones críticas o complejas
- En caso que se requiera, no es posible recurrir a jornadas laborales extra de desarrollo (es muy usual), ya que no se cuenta con personal interno para ejercer esta labor y no se tiene poder sobre el proveedor

Según [HSC97], existen principalmente tres grupos de participantes en un proyecto de desarrollo de software:

- *El cliente:* como ya se dijo, el cliente hace referencia a la organización contratante y su principal expectativa tiene que ver con la obtención de un producto completamente funcional en el entorno objetivo. Adicionalmente, el cliente desea minimizar los costos del proyecto, evitar a toda costa retrasos, maximizar y agilizar el retorno sobre la inversión, y manejar una relación saludable con el proveedor.
- *El usuario final:* el usuario final, al igual que el cliente, tiene como principal objetivo obtener un producto completamente operativo en el menor tiempo posible.
- *El proveedor:* el proveedor hace referencia a la organización contratista y su principal objetivo es la maximización de sus ganancias. Adicionalmente, el proveedor quisiera asegurar una fuente futura de ingresos a través de la consecución de nuevos contratos.

La anterior explicación tiene un objetivo: mostrar que, como en el Cálculo, los proyectos de adquisición de software son un problema de optimización, donde

---

<sup>15</sup> [SVE01], p. 10.

las partes definen intereses que entran en conflicto [SVE01]. Prueba de lo anterior es, por ejemplo, el deseo del proveedor de maximizar sus ganancias contra la necesidad del cliente de minimizar los costos del proyecto, o la pretensión del cliente de maximizar el desempeño técnico del producto software a la vez que los plazos de los ciclos de desarrollo se minimizan. La tabla 4.14 sintetiza las principales expectativas de las participantes de un proyecto de desarrollo de software.

<b>Participante</b>	<b>Objetivos / Expectativas / Deseos</b>
<i>Usuario Final</i>	<ul style="list-style-type: none"> <li>• Maximizar el desempeño técnico del producto</li> <li>• Minimizar los tiempos de desarrollo</li> </ul>
<i>Cliente</i>	<ul style="list-style-type: none"> <li>• Maximizar el desempeño técnico del producto</li> <li>• Maximizar (mejorar, mantener) la relación con el proveedor</li> <li>• Maximizar y agilizar el retorno de la inversión</li> <li>• Minimizar costos</li> <li>• Minimizar plazos de desarrollo y retrasos</li> </ul>
<i>Proveedor</i>	<ul style="list-style-type: none"> <li>• Maximizar las ganancias</li> <li>• Maximizar las fuentes de ingresos futuras</li> </ul>

Tabla 4.14. Objetivos de las participantes de un proyecto de desarrollo de software. Basado en [SVE01].

Por último, es claro que con el fin que un contrato se vuelva operativo, es decir, para que entre en ejecución, ambas o alguna de las partes involucradas tienen que ceder o sacrificar alguna pretensión. Por ejemplo, si el cliente requiriera realmente minimizar los costos del proyecto, este tendría que sacrificar la pretensión de maximizar el desempeño técnico del producto porque esto implica más trabajo, más tiempo y por lo tanto más recursos financieros.

#### **4.8.2. Escenarios de un Proyecto de Desarrollo de Software**

En los proyectos de desarrollo de software se pueden presentar varios escenarios. El más común y el más simple, es cuando en un proyecto solo existen dos partes involucradas: un cliente y un proveedor. Este escenario, aunque ideal, no siempre es el caso, ya que entre más complejo sea el

proyecto mas son los recursos involucrados y mas complejas se vuelven las relaciones. A continuación se sintetiza una clasificación que describe estos escenarios [GAM99]:

- *Relación simple*: en el proyecto solo existen dos partes; un proveedor y un cliente.
- *Múltiples proveedores*: se refiere a los proyectos de desarrollo donde participan dos o más proveedores y un solo cliente.
- *Tercerización conjunta*: la tercerización conjunta se refiere a cuando dos o más organizaciones cliente contratan a un solo proveedor para desarrollar el proyecto.
- *Relaciones complejas*: esta clasificación es una combinación de múltiples proveedores con la tercerización conjunta, es decir, cuando varias organizaciones cliente se ven involucradas con varios proveedores en un proyecto de desarrollo de software.

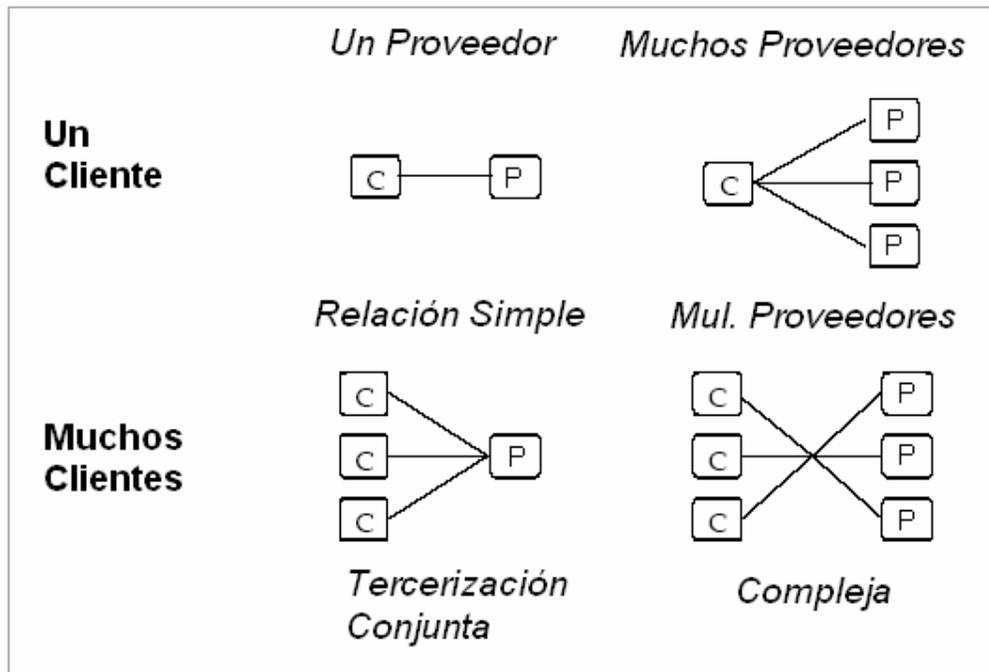


Figura 4.11. Escenarios de un proyecto de desarrollo de software. Adaptado de [SVE01].

La clasificación presentada es adecuada porque de ella se pueden extraer conclusiones importantes en el contexto de los proyectos delegados de desarrollo de software. La primera de ellas tiene que ver con el hecho que, según lo mostrado, se podría pensar que el escenario mas complicado es el de las relaciones complejas. De manera casi siempre indeseable, un escenario como el mencionado se podría complicar aún mas: los proveedores podrían establecer relación con otras organizaciones externas para apoyarlos en el desarrollo de algunas de las funciones contratadas con el cliente; esta situación, en proyectos de desarrollo de software, casi nunca es conveniente para el cliente. Como segundo, se puede apreciar que un escenario como el de múltiples vendedores puede llegar a ser muy conveniente para el cliente: este escenario promueve la especialización de los proveedores y reduce el riesgo en cuanto a la asunción de actitudes oportunistas por parte de estos, dado que cualquier otro puede entrar a reemplazarlo sin mayores inconvenientes. Por último, puede decirse que un escenario como el de la tercerización conjunta ayuda a diversificar el riesgo que se presenta cuando un cliente por si solo contrata con un proveedor. No obstante, la tercerización conjunta puede dificultar ostensiblemente la coordinación que obligatoriamente se requiere entre las organizaciones cliente (lo cual puede redundar en mayores costos) y puede maximizar el riesgo que se difunda información privilegiada, dada la misma naturaleza de esta tipo de escenario [GAM99].

#### **4.8.3. Problemática de la Delegación**

Según Salwin <sup>16</sup> los problemas que caracterizan las adquisiciones que involucran una cantidad significativa de desarrollo de software son:

- Los productos software no son terminados a tiempo
- Los costos exceden lo presupuestado
- Los resultados alcanzados no reúnen las expectativas planteadas

---

<sup>16</sup> [SAA98], v. 1, p. 1.

De manera indeseable, estas características además de ser un riesgo latente en todo proyecto de desarrollo de software, muy frecuentemente se convierten en realidad. En este trabajo, como en muchos otros, se acude nuevamente al tristemente célebre “Reporte del Caos” para contextualizar la problemática del desarrollo delagado. El reporte del caos fue publicado en 1995 por el Standish Group [FAR06], una organización que fundamentalmente se dedica a realizar investigaciones sobre todo lo que tiene que ver con las Tecnologías de la Información. Básicamente, fueron investigados 8380 proyectos de desarrollo de software para determinar cuán comunes eran los problemas arriba mencionados. Para asegurar la confiabilidad de los resultados, se tomaron muestras representativas de los sectores mas importantes de la industria, que incluían tanto pequeñas, medianas, como grandes compañías. Los sectores incluídos fueron: bancario, salud, manufactura, seguridad, ventas al por menor, ventas al por mayor, aseguradoras, servicios y organizaciones gubernamentales tanto locales, estatales, como federales. A pesar que el reporte no expone directamente indicadores acerca del grado de tercerización involucrado en estos proyectos, los resultados obtenidos dan una clara idea de estos problemas:

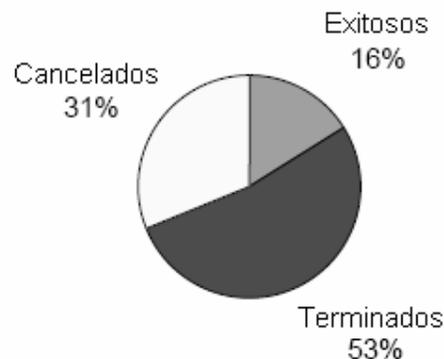


Figura 4.12. Resultados descritos en el reporte del caos. Fuente [STG95].

- El 16% de los proyectos fueron terminados a tiempo y no se sobrepasaron del presupuesto asignado; las soluciones desarrolladas

quedaron 100% funcionales con todas las características que fueron especificadas (*proyectos exitosos*).

- El 53% de los proyectos fueron terminados y las soluciones desarrolladas quedaron operativas, pero dichas soluciones ofrecían menos características de las que originalmente fueron especificadas, además que se presentaron sobrecostos y serios retrasos (*proyectos terminados*).
- El 31% de los proyectos fueron cancelados en algún punto del ciclo de desarrollo (*proyectos cancelados*).
- El proyecto promedio excedía el presupuesto en un 90% y los plazos establecidos en un 120%.

<b>Factores que Perjudicaron los Proyectos</b>	<b>% de Respuestas</b>
Requisitos Incompletos	13,10%
Falta de Involucramiento de los Usuarios	12,40%
Carencia de Recursos	10,60%
Expectativas poco Realistas	9,90%
Abandono de los Directivos	9,30%
Modificación de los Requerimientos y las Especificaciones	8,70%
Falta de Planeación	8,10%
No se pudo Prolongar Mas	7,50%
Carencia de Gestión TI	6,20%
Analfabetismo Tecnológico	4,30%
Otros	9,90%

Tabla 4.15. Razones para el fracaso de los proyectos. Fuente [STG95].

Como puede observarse los resultados son alarmantes y, como la investigación no hizo distinción entre los proyectos de desarrollo internos y los proyectos de desarrollo delegados, estos últimos se ven cubiertos por ellos. La pregunta mas obvia en este punto sería ¿qué causo que estos proyectos fracasaran? La respuesta se encuentra contenida en la tabla 4.15 y los resultados también son fruto de la investigación del Standish Group. Como se puede observar en la tabla, el factor que más influyó para que estos proyectos fracasaran, tiene que ver con una pobre especificación de los requisitos que el producto que se esperaba debía reunir. Sin embargo, existen opiniones ajenas al estudio que focalizan las causas de estos fracasos en otros factores. Según [BRO98], la

causa principal de los problemas presentados en la adquisición de software no proviene de dificultades técnicas; mas bien, tiene que ver con una pobre gestión que, dado el encarecimiento y la creciente complejidad del software, hará que los problemas evidenciados en el estudio se hagan aún mas graves. Infortunadamente, y como menciona Salwin<sup>17</sup>, no siempre estos problemas se detectan al inicio de los proyectos lo cual ensombrece aún mas la situación. A continuación se brindan algunos comentarios acerca de los resultados contenidos en la tabla 4.15 y algunas apreciaciones del autor:

- *Requisitos incompletos*: los clientes y los usuarios dan un conjunto incompleto y no validado de requerimientos al proveedor y el proveedor no se preocupa por elicitarlos con mayor rigurosidad<sup>18</sup>.
- *Falta de involucramiento de los usuarios*: no se tiene un usuario líder que aglutine las ideas del grupo de usuarios y de los clientes, y que incorpore el punto de vista de los mismos acerca de la funcionalidad y la usabilidad del producto.
- *Prometer en exceso*: el personal comercial del proveedor muchas veces hace promesas que el equipo delegado de desarrollo no puede implementar o que son técnicamente inviables
- *Modificación de los requerimientos*: mientras que el proyecto se encuentra en curso, el cliente continúa modificando el alcance y añadiendo nuevas funcionalidades, y no tiene en cuenta que el cronograma es apretado y que los recursos son limitados.
- *Fragmentación*: tanto el cliente como el proveedor están involucrados en otros proyectos, a los cuales, sin ninguna consideración por el proyecto que los relaciona, sin orden o planificación, y sin ningún tipo de aviso, dedican excesivo tiempo.

---

<sup>17</sup> [SAA98], v. 1, p. 1.

<sup>18</sup> [SOS97] define la elicitación de requisitos como sistemáticamente extraer e inventariar los requisitos de un sistema de una combinación de actores humanos, del entorno del sistema, de estudios de factibilidad, del análisis del mercado, de los planes de negocio, y del análisis de productos competidores y del domino del conocimiento.

- *Expectativas poco realistas*: es muy común en los proyectos de desarrollo de software, que se planteen cronogramas imposibles de cumplir, que se abuse o no se sea conciente de las restricciones existentes en cuanto a recursos se refiere, y que no se tengan en cuenta las limitaciones que hasta las tecnologías mas novedosas imponen.
- *Carencia de recursos*: muchas veces se presenta que el proyecto no tiene una base financiera sólida o no cuenta con el suficiente patrocinio. Adicionalmente, se puede carecer del personal, las herramientas, los equipos y demás elementos requeridos para desarrollar el proyecto.
- *Abandono de los directivos*: es muy común, que los directivos de la organización cliente por algún motivo no tengan interés en el proyecto que se esta desarrollando. Bajo estas circunstancias, difícilmente el proyecto pueda fructificar.
- *Carencia de competencias*: los participantes del proyecto no cuentan con las suficientes habilidades técnicas, humanas y de liderazgo, para poder enfrentar los retos que el proyecto propone.
- *Fricción*: los proyectos se pueden ver afectados porque algunos de los miembros del equipo, por algún motivo, no se llevan bien.
- *Comunicaciones inefectivas*: este factor es determinante para cualquier tipo de proyecto. Si se carece de canales efectivos de comunicación, y si la información no llega a quién tiene que llegar de una manera oportuna, el proyecto se encuentra condenado al fracaso.
- *Objetivos poco claros*: los objetivos de un proyecto de desarrollo, determinan el norte que el proyecto debe seguir y las actividades requeridas para alcanzarlos. La falta de claridad en cuanto a objetivos se refiere y de una visión de proyecto bien elaborada, hace que los participantes vayan en direcciones distintas y muchas veces contrarias.
- *Falta de disciplina*: no se tienen establecidos procesos formales de desarrollo o los existentes son letra muerta, es decir, no son seguidos.
- *Carga administrativa*: se realiza demasiada gestión administrativa para verificar que los acuerdos contractuales están siendo cumplidos, y los

aspectos técnicos que son los que finalmente agregan valor son relegados o tratados con laxitud.

- *Inexistencia de indicadores*: muchas veces las directivas del proyecto no disponen de indicadores que les permitan establecer con certeza el estado del mismo. Adicionalmente, sucede con frecuencia que los pocos indicadores presentados no son confiables, y desempeñar con efectividad las labores de monitoreo y control de un proyecto bajo estas circunstancias es casi imposible.
- *No existe un líder*: muchas veces los proyectos carecen de una cabeza visible que gestione y canalice todo lo que tenga que ver con los mismos. Lo anterior, con el agravante que la organización del proyecto define demasiados roles de dirección (“muchos jefes y pocos indios”), lo cual burocratiza e impide la toma oportuna de decisiones.
- *Incompetencia en la adquisición*: muchas veces se yerra en canalizar, gestionar y entender las necesidades y requerimientos particulares de un proyecto delegado de desarrollo de software.

Como una consideración final, es preciso anotar que lamentablemente en el contexto nacional no existe una entidad u organización visible que se encargue de investigar el entorno de la industria del software en el país. Lo anterior, naturalmente, dificulta la recolección de datos lo suficientemente fundamentados y confiables como para establecer el estado del arte del desarrollo delegado de software en Colombia.

## 5. MARCO DE PROCESOS PARA EL PROCESO DELEGADO DE DESARROLLO DE SOFTWARE

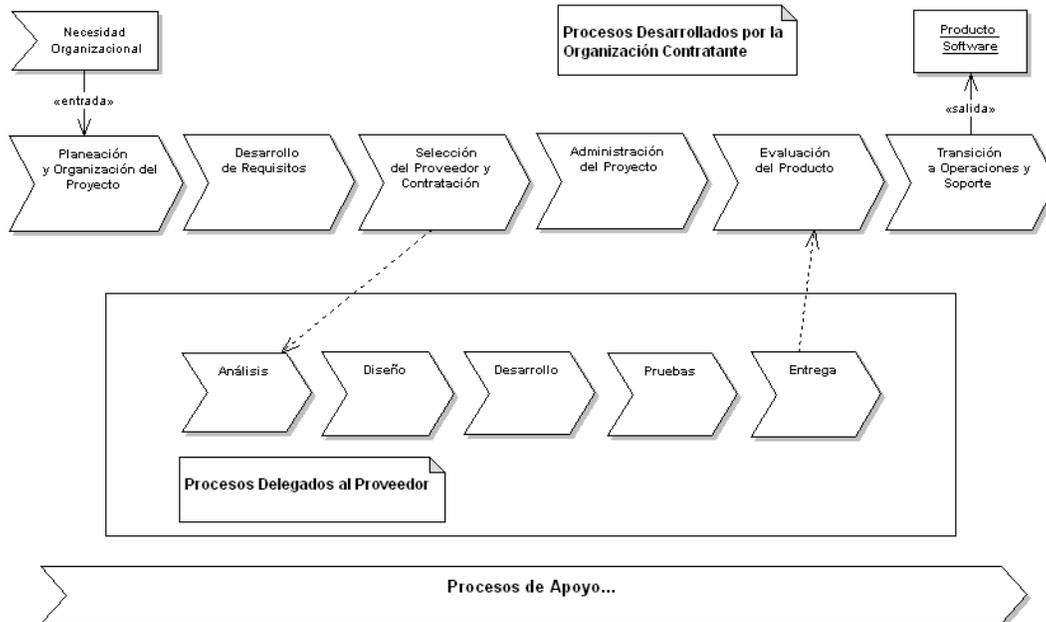


Figura 5.1. Marco de procesos para el Proceso Delegado de Desarrollo de Software<sup>19</sup>.

La figura 5.1 resume la propuesta desarrollada en este proyecto. Básicamente, se identifican los procesos clave que intervienen en un proyecto donde se delega en el proveedor la responsabilidad de desarrollar el producto software. Estos procesos, a los cuales se denominará procesos de negocio, se dividen en tres grandes grupos: los procesos ejecutados por el proveedor, los procesos de apoyo y los procesos desempeñados por la organización cliente. Los procesos mostrados en la parte baja de la figura representan a un ciclo de vida clásico del desarrollo de software donde las actividades son llevadas a cabo de una manera secuencial<sup>20</sup>. Dichos procesos son los que se delegan en el

<sup>19</sup> Basado en: [GAL06], p. 10.

<sup>20</sup> El modelo secuencial, o modelo en cascada, fue inicialmente propuesto por [ROY70] en el clásico “Managing the Development of Large Software Systems: Concepts and Techniques”.

proveedor y contribuyen de manera directa a la construcción del producto software; esto son:

- Análisis
- Diseño
- Desarrollo
- Pruebas
- Entrega

Cabe anotar, que se hace referencia a este ciclo de vida única y exclusivamente para propósitos de ilustración, que las fases pueden tomar otros nombres, y que en el la actualidad las organizaciones modernas que se dedican al desarrollo de software emplean técnicas o metodologías de desarrollo que introducen severas variaciones sobre este modelo.

Los procesos de apoyo, como su nombre lo indica, son procesos que contribuyen a la correcta realización de los procesos de adquisición y/o actúan como suplemento ideal de los mismos. Como se describió en el marco teórico de este trabajo (ver literal CMMI), si se recurre a los modelos CMMI estos procesos podrían hacer referencia a los procesos de Gestión de la Configuración, Aseguramiento de la Calidad de Proceso y Producto, Medición y Análisis, entre otros. Estos procesos no son incluidos dentro del alcance de este proyecto, pero se supone que como apoyo o complemento a la estrategia propuesta debe recurrirse a ellos, en especial al proceso de la Gestión de la Configuración.

Por último, y bajo el contexto del desarrollo delegado de software, se encuentran los procesos desempeñados por la organización contratante. Estos procesos son descritos en la parte superior de la figura 5.1 y se les denomina, según esta propuesta, procesos de adquisición. Los procesos adquisición son los que bajo la óptica de las organizaciones cuyo objeto de negocio no es el de

desarrollar software deberían desempeñarse, y se convierten en el ciclo de vida del software que mejor se adecua al entorno en el que estas se mueven. Cabe anotar, que el marco de procesos propuesto (procesos de adquisición) asume que las actividades preliminares al inicio de cualquier proyecto tales como el análisis de viabilidad, la consecución del patrocinio, la decisión de obtener un producto software hecho a la medida y una planeación preliminar, ya han sido desarrolladas. Los procesos que se trabajan en el marco de procesos son:

### *Planeación y Organización del Proyecto*

Este proceso tiene como propósito la personalización de los procesos de adquisición de acuerdo a las necesidades particulares del proyecto y la creación de un plan de adquisición que guíe de manera adecuada dichos procesos.

### *Desarrollo de Requisitos*

El propósito del proceso de Desarrollo de Requisitos es establecer, clarificar y definir el conjunto de requisitos que el producto software debe satisfacer, y que estos puedan ser entendidos tanto por la organización contratante, como por el proveedor y los participantes del proyecto.

### *Selección del Proveedor y Contratación*

El propósito del proceso de Selección del Proveedor y Contratación es desarrollar una solicitud de propuesta que identifique y aglutine las necesidades organizacionales respecto a la adquisición del producto software,

y seleccionar el proveedor que mejor pueda satisfacer los requerimientos contractuales<sup>21</sup>.

### *Administración del Proyecto*

El propósito del proceso de Administración del Proyecto es monitorear el desempeño del proveedor, obtener información acerca del estado del proyecto, resolver problemas, administrar los pagos, y tratar con otros temas contractuales.

### *Evaluación del Producto*

El propósito del proceso de evaluación del producto<sup>22</sup> es verificar que los entregables generados por el proveedor reúnen los requerimientos especificados y los objetivos de calidad.

### *Transición a Operaciones y Soporte*

El propósito del proceso de transición para el soporte, es la minimización de los traumatismos generados por la transición de los entregables del proyecto a la organización de soporte y mantenimiento y verificar que dichos entregables y otros servicios, como capacitación a los usuarios e instalación del producto software, han sido recibidos a satisfacción.

---

<sup>21</sup> [COF02], p. 30.

<sup>22</sup> Cuando en este trabajo explícitamente no se hace la aclaración del término producto a producto software, este último puede ser entendido como cualquier salida tangible o servicio que es el resultado de un proceso y cuyo propósito es la entrega a un cliente o usuario final. Un producto es un producto de trabajo que es entregado al cliente. [CKS03], p. 624.

Respecto a la composición de los procesos, cada proceso de adquisición define una estructura en la cual se incluyen objetivos, propósito, actividades a desarrollar, tareas a realizar y la descripción global de los entregables mas representativos que son generados. De igual modo, durante el desarrollo de la estrategia planteada se incluyen una serie de listas de chequeo cuyo propósito es asegurar que las actividades mas relevantes de cada proceso son desarrolladas y que los elementos clave de cada entregable son incluidos; estas listas de chequeo son:

- Especificación de Requisitos
- Plan de Pruebas de Aceptación
- Procedimiento de Prueba
- Caso de Prueba
- Registro de Prueba
- Reporte de Pruebas
- Cronograma del Proyecto
- Habilidades del Equipo de Soporte
- Roles Requeridos para el Soporte del Producto
- Plan de Mantenimiento
- Derechos de Propiedad
- Solicitud de Propuesta

Todo lo anterior hace parte de la estrategia propuesta, la cual recoge y sintetiza elementos de modelos de calidad internacionalmente reconocidos como CMMI, SA-CMM, CMMI-AM, y de metodologías como el Proceso Unificado de Desarrollo de Software<sup>23</sup>. Esta estrategia, denominada *Marco de Procesos para el Proceso Delegado de Desarrollo de Software*, apunta al desarrollo de unos procesos de adquisición usables que permitan la obtención de productos software de una mejor calidad, y supone que el seguimiento juicioso de las

---

<sup>23</sup> Tras haber evolucionado, el Proceso Unificado de Desarrollo de Software es mas conocido hoy en día como RUP [JBR00].

actividades, tareas, entregables y listas de chequeo planteadas, presume un sistema de aseguramiento de la calidad lo suficientemente completo como para gestionar eficiente y eficazmente proyectos donde la responsabilidad del proceso de desarrollo se traslada a un proveedor.

### 5.1. Planeación y Organización del Proyecto

Este proceso tiene como propósito la personalización de los procesos de adquisición de acuerdo a las necesidades particulares del proyecto y la creación de un plan de adquisición que guíe de manera adecuada dichos procesos.

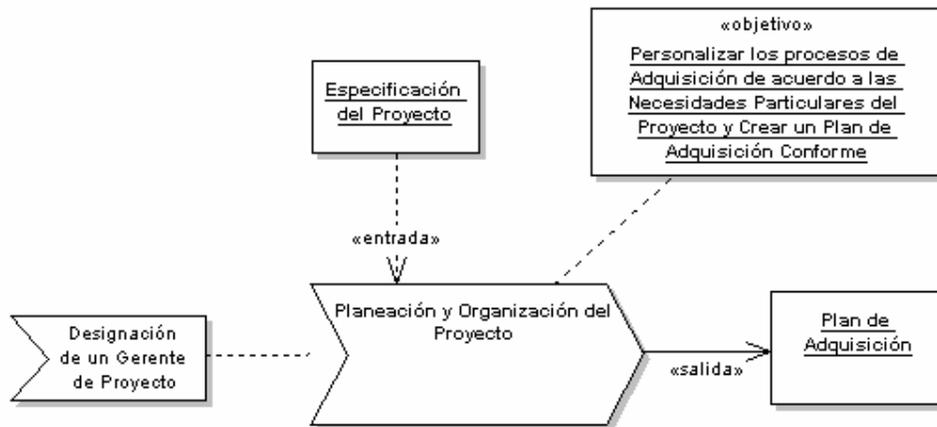


Figura 5.2. Diagrama de proceso de negocio del proceso Planeación y Organización del Proyecto.

La figura 5.2 muestra los elementos esenciales que intervienen en proceso de Planeación y Organización del Proyecto. El proceso inicia cuando de manera formal se designa un Gerente de Proyecto, el cual tiene la responsabilidad de hacer la planeación preliminar, asignar responsabilidades y recursos, y de asegurarse que el proyecto de delegación arranque de manera adecuada. La Planeación y Organización del Proyecto se alimenta de las especificaciones globales del proyecto y, tras llevadas a cabo las tareas que define, se obtiene

un plan de adquisición que supone una guía adecuada para hacer gestión del mismo.

### **5.1.1. Objetivos**

- Seleccionar y configurar los procesos de adquisición de acuerdo a las necesidades y características particulares del proyecto
- Identificar, seleccionar y asignar a los procesos de adquisición el personal adecuado
- Prever y documentar en un plan todos los recursos, tiempos, costos, especificaciones, y demás variables que deben ser contempladas en un proyecto de adquisición

### **5.1.2. Actividades a Desarrollar**

- Personalizar los Procesos de Adquisición
- Conformar Grupos para todos los Procesos de Adquisición
- Desarrollar el Plan de Adquisición
- Revisar el Plan de Adquisición y Aprobar

#### **5.1.2.1. Personalizar los Procesos de Adquisición**

Como es de esperar, la forma en que se trabaje un proyecto depende de las características propias del mismo. En este sentido, es importante establecer un nivel adecuado de personalización para los procesos, actividades, roles y entregables de acuerdo a las necesidades particulares de cada proyecto. Para este propósito, es muy recomendable acudir a la experiencia obtenida en otros proyectos de adquisición similares, seleccionar los métodos de trabajos apropiados y adecuarlos a las características propias del proyecto. El insumo principal de esta actividad es la especificación del proyecto, en la cual se

encuentran contenidas las características y necesidades propias del mismo. A continuación se describen algunos de los elementos que deben tenerse en cuenta a la hora de realizar la personalización.

#### 5.1.2.1.1. Nivel de Formalidad del Proyecto

Para que el proyecto sea efectivo, es necesario establecer el nivel de formalidad con que se trabajarán los procesos de adquisición definidos de acuerdo a las necesidades propias del mismo. Básicamente, se pueden distinguir tres niveles de formalidad:

- Procesos livianos: el proyecto puede ser administrado fácilmente y los objetivos de los procesos establecidos son alcanzados con un nivel de rigidez bajo. En los procesos se identifican y trabajan las actividades absolutamente necesarias para que el proyecto marche sin problemas. Este tipo de personalización es adecuada para proyectos de bajo riesgo.
- Procesos controlados: Este nivel de formalidad es adecuado para proyectos relativamente complejos y en los cuales existe algún riesgo que las cosas se salgan de control. Además de las actividades absolutamente necesarias, se identifican y desarrollan actividades cuyo propósito es ejercer mayor control sobre el proyecto. Este nivel de formalidad es adecuado para proyectos de riesgo medio.
- Procesos robustos: Este tipo de nivel de formalidad es utilizada para proyectos de alto riesgo, donde el nivel de complejidad es bastante elevado y donde existe una fuerte tendencia a caer en una situación de caos a menos que este se maneje con una férrea disciplina. En los procesos se introducen actividades que incrementan el nivel de formalidad del proyecto y que hacen énfasis en las labores administrativas. Es de anotar, que la rigidez del proyecto es claramente aumentada y que esto tiene implicaciones en su eficiencia; en contra posición, se obtiene control efectivo del mismo y unas salidas más confiables.

#### 5.1.2.1.2. Factores que Influyen en la Personalización

A continuación se describen brevemente algunos de los factores que deben tenerse en cuenta a la hora de personalizar los procesos de adquisición.

##### 5.1.2.1.2.1. Características del Producto

Estos factores tienen que ver con las características del producto software a ser desarrollado.

###### 5.1.2.1.2.1.1. Estabilidad de los Requisitos

Debido al trato artesanal que por naturaleza se le ha dado al proceso de desarrollo de software y a la incapacidad de anticipar todos los problemas técnicos que se puedan presentar, es muy probable que se redefinan requisitos durante el proceso. Sin embargo, dependiendo de la complejidad del problema a ser resuelto a través del desarrollo del producto software, las modificaciones en la definición de los requisitos pueden presentarse con mayor o menor frecuencia. La tabla 5.1 define el nivel de formalidad requerido en la administración de requisitos de acuerdo a su estabilidad:

<b>Formalidad del Proceso</b>	<b>Observaciones</b>
Liviano	Los requisitos son estables y existe una baja probabilidad que puedan ser modificados. El problema a ser resuelto es poco complejo.
Controlado	El problema a ser resuelto es de complejidad media y un número relativamente considerable de requisitos podrían ser redefinidos.
Robusto	El problema a ser resuelto es de complejidad alta y se tiene incertidumbre acerca de la estabilidad de muchos de los requisitos definidos.

Tabla 5.1. Formalidad de los procesos vs. estabilidad de los requisitos. Basado en [SVE01].

#### 5.1.2.1.2.1.2. *Tamaño del Producto Software*

El tamaño del producto software es una buena métrica que indica el esfuerzo requerido para desarrollarlo. En la literatura pueden encontrarse muchas técnicas para medir el tamaño de un producto software (por ejemplo, la Estimación Basada en Casos de Uso [PER04] pero para efectos prácticos se utilizará una de las más tradicionales: miles de líneas de código. La tabla 5.2 define el nivel de formalidad requerido en el proceso de adquisición de acuerdo al tamaño del producto software a desarrollar. Se debe tener en cuenta que los valores definidos en la tabla son relativos ya que dependen de factores como el lenguaje de programación y la metodología de programación:

<b>Formalidad del Proceso</b>	<b>Tamaño del Producto</b>
Liviano	Pequeño. Menos de 32.000 líneas de código.
Controlado	Mediano. Entre 32.000 y 128.000 líneas de código.
Robusto	Grande. Más de 128.000 líneas de código.

Tabla 5.2. Formalidad de los procesos vs. tamaño del producto software. Basado en [SVE01].

#### 5.1.2.1.2.1.3. *Criticidad*

Según [OSG96], la criticidad de un producto software tiene que ver con el impacto que este tiene en la seguridad de las personas y en la magnitud de las pérdidas financieras que pudieran presentarse en caso de un mal funcionamiento en el mismo. La tabla 5.3 esboza el nivel de formalidad requerido en el proceso de adquisición de acuerdo a la criticidad del producto software a desarrollar:

<b>Formalidad del Proceso</b>	<b>Criticidad del Producto</b>
Liviano	La seguridad de las personas no se pone en riesgo y un mal funcionamiento del producto no genera pérdidas financieras.

Tabla 5.3. Formalidad de los procesos vs. criticidad del producto software. Basado en [SVE01].

Formalidad del Proceso	Criticidad del Producto
Controlado	La seguridad de las personas no se pone en riesgo pero existe la probabilidad que un mal funcionamiento del producto genera pérdidas financieras.
Robusto	Si el producto software falla o presenta un mal funcionamiento, la seguridad de las personas corre grave riesgo o muy probablemente se incurra en grandes pérdidas financieras.

Tabla 5.3. Formalidad de los procesos vs. criticidad del producto software. Basado en [SVE01].

#### 5.1.2.1.2.1.4. Innovación

La innovación tiene que ver con la aparición de problemas de los cuales no se tiene alguna referencia anterior o cuyas características difieren enormemente de problemas anteriormente resueltos. Como ingrediente adicional, se tiene que no se cuenta con evidencia acerca que el problema haya sido resuelto a través de alguna estrategia computarizada. Los proyectos menos innovadores hacen uso de tecnologías muy maduras y ampliamente probadas y, por tanto, requieren de procesos más livianos.

Formalidad del Proceso	Innovación del Proyecto
Liviano	Innovación baja. Problema ampliamente conocido. Existencia de muchos productos software que lo resuelven. Utilización de tecnologías maduras.
Controlado	La tecnología a utilizar ha probado funcionar pero existen pocas aplicaciones que la implementan.
Robusto	La tecnología a utilizar es emergente y no se ha estandarizado completamente. Se tiene incertidumbre acerca de si el problema puede ser resuelto a través de alguna estrategia computarizada.

Tabla 5.4. Formalidad de los procesos vs. innovación del proyecto. Basado en [SVE01].

#### 5.1.2.1.2.1.5. *Impacto Organizacional*

El impacto organizacional tiene que ver con el impacto que el producto software tiene sobre los procesos organizacionales, la forma en que el usuario percibe y desempeña su trabajo y sobre otros productos de la organización contratante.

<b>Formalidad del Proceso</b>	<b>Impacto Organizacional</b>
Liviano	El producto software tiene un impacto mínimo en la forma en que el usuario percibe y desempeña su trabajo.
Controlado	Se introducen cambios moderados en la forma en que el usuario realiza su trabajo, en los procesos organizacionales y/o en otros productos.
Robusto	Se introducen cambios drásticos en la forma en que el usuario realiza su trabajo, en los procesos organizacionales y/o en otros productos.

Tabla 5.5. Formalidad de los procesos vs. impacto organizacional. Basado en [SVE01].

#### 5.1.2.1.2.2. Esfuerzo

Los factores asociados al esfuerzo tienen que ver con la cantidad de trabajo y los recursos requeridos para desarrollar el producto software. El tamaño del equipo de desarrollo, el tiempo requerido para liberar el producto y el costo total del mismo, son ejemplos de métricas utilizadas para estimar el esfuerzo. Entre mayor sea el esfuerzo requerido más formales deben ser los procesos de adquisición implicados.

##### 5.1.2.1.2.2.1. *Tiempo de Liberación del Producto*

El tiempo de liberación del producto es la fecha estimada para la puesta en productivo del producto software terminado. Se debe tener en cuenta que para proyectos grandes o de alta complejidad, existirá una mayor probabilidad que nuevo personal entre a engrosar el equipo del proyecto y que algunos otros lo dejen. De igual manera, bajo las condiciones mencionadas, es muy probable que se hagan modificaciones sobre los requisitos inicialmente definidos. La

tabla 5.6 describe el nivel de formalidad requerido en el proceso de adquisición de acuerdo a la fecha de liberación del producto estimada. Debe tenerse en cuenta que los valores allí sugeridos pueden variar de acuerdo a las particularidades de cada proyecto:

<b>Formalidad del Proceso</b>	<b>Tiempo de Liberación del Producto</b>
Liviano	Menos de 18 meses.
Controlado	Entre 12 y 36 meses.
Robusto	Mayor a 24 meses.

Tabla 5.6. Formalidad de los procesos vs. tiempo de liberación del producto. Basado en [SVE01].

#### 5.1.2.1.2.2.2. *Costo Total*

El costo total del producto se refiere a la sumatoria de todos los costos implicados en el proceso de adquisición que van desde la concepción del proyecto hasta la puesta en producción del producto software: costos del personal interno que participa en el proyecto, costos del proveedor, costos administrativos implicados y similares. La tabla 5.7 describe el nivel de formalidad requerido en el proceso de adquisición de acuerdo al costo total involucrado. Dicho costos se expresan en términos variables dada la particular percepción que tenga cada compañía en cuanto a lo que es una alta o baja inversión.

<b>Formalidad del Proceso</b>	<b>Costo Total del Producto</b>
Liviano	Bajo. Menos de \$ $X$ .
Controlado	Medio. Entre \$ $X$ y \$ $Y$ ( $Y > X$ )
Robusto	Alto. Mayor que \$ $Y$ .

Tabla 5.7. Formalidad de los procesos vs. costo total del producto. Basado en [SVE01].

### 5.1.2.1.2.2.3. *Tamaño del Equipo de Desarrollo del Proveedor*

Esta métrica se refiere al número de personas de la empresa proveedora involucradas de manera directa en las actividades de desarrollo de software. Entre más grande sea este número más disciplina se debe tener con las estrategias de comunicación establecidas. Lo anterior tiene como propósito facilitar la coordinación efectiva y eficiente del trabajo, sortear malentendidos y resolver conflictos. La tabla 5.8 describe el nivel de formalidad requerido en el proceso de adquisición de acuerdo al número de personas involucradas en las labores de desarrollo. Al igual que con las métricas anteriores, debe tenerse en cuenta que este número no es absoluto ya que se depende de las características propias de cada proyecto.

<b>Formalidad del Proceso</b>	<b>Tamaño del Equipo</b>
Liviano	Pequeño. Menos de 10 desarrolladores.
Controlado	Mediano. Entre 10 y 30 desarrolladores.
Robusto	Grande. Más de 30 desarrolladores.

Tabla 5.8. Formalidad de los procesos vs. tamaño del equipo de desarrollo. Basado en [SVE01].

### 5.1.2.1.2.3. Entorno de Contratación

Los factores del entorno de contratación son aquellos que tienen que ver directamente con el proveedor y con el entorno organizacional en el cual el proceso de adquisición tomará lugar. Elementos como el número de organizaciones involucradas en el proceso de adquisición, las habilidades del “equipo delegado” de desarrollo, la dispersión geográfica y la experiencia previa con el contratista, son ejemplos de factores que afectan el entorno de contratación.

#### 5.1.2.1.2.3.1. Número de Organizaciones Involucradas

Bajo el contexto de esta métrica, el término organización hace referencia a cualquier entidad externa que sea participante del proyecto de adquisición y que tenga responsabilidades dentro del mismo. Proveedores que soporten sistemas, consultores y empresas de testing son ejemplos de este tipo de entidades. Al igual que con el “equipo delegado” de desarrollo de software, entre más grande sea el número de empresas contratistas involucradas en el proyecto, más disciplina se debe tener con la comunicación para coordinar efectivamente las tareas delegadas a estos.

<b>Formalidad del Proceso</b>	<b>Organizaciones Involucradas en el Proyecto</b>
Liviano	Relación estrecha entre dos partes: organización contratante y proveedor.
Controlado	Más de dos partes involucradas: organización contratante y terceros que utilizan outsourcing (se convierten en clientes); o la organización contratante trabaja en forma directa con varios proveedores. La estructura organizacional implícita no es muy compleja.
Robusto	Múltiples organizaciones trabajan bajo una estructura organizacional compleja la cual puede contener: múltiples clientes (organización contratante y terceros que utilizan terceros), múltiples proveedores, múltiples subcontratistas, múltiples contratistas de soporte o cualquier otra combinación.

Tabla 5.9. Formalidad de los procesos vs. número de organizaciones involucradas en el proyecto. Basado en [SVE01].

#### 5.1.2.1.2.3.2. Características del Equipo Desarrollador del Proveedor

Elementos como la experiencia y la experticia, el desempeño y la constitución, son factores determinantes del equipo desarrollador. La tabla 5.10 describe el nivel de formalidad requerido en el proceso de adquisición de acuerdo a las características mencionadas.

Formalidad del Proceso	Características Equipo Desarrollador
Liviano	Equipo con experiencia y bien entrenado. Alto desempeño. Baja rotación del personal. Métodos apropiados y eficientes de trabajo. Buen trabajo en equipo.
Controlado	Algunos integrantes no tienen experiencia. El trabajo en equipo presenta algunas dificultades. Rotación de personal moderada. Los métodos de trabajo necesitan ser ajustados. Desempeño medio.
Robusto	Alta rotación de personal. La gran mayoría de los integrantes del equipo son inexpertos. Se presentan muchos inconvenientes con el trabajo en equipo. Los métodos de trabajo son ineficientes. El desempeño es bajo.

Tabla 5.10. Formalidad de los procesos vs. características del equipo delegado de desarrollo. Basado en [SVE01].

#### 5.1.2.1.2.3.3. *Experiencia con el Proveedor*

En términos de la curva de aprendizaje del proveedor, del conocimiento del personal indicado y del acoplamiento a las prácticas y entorno organizacionales, las experiencias previas que se hayan tenido con un proveedor pueden ser muy importantes. La tabla 5.11 describe el nivel de formalidad requerido en el proceso de adquisición de acuerdo a la experiencia que se haya tenido con el proveedor:

Formalidad del Proceso	Experiencia Previa con el Proveedor
Liviano	La organización contratante y el proveedor han trabajado varias veces en proyectos anteriores los cuales han sido exitosos. Se ha construido una relación basada en la colaboración y la confianza. Ambas partes conocen bien como trabajan.
Controlado	El proveedor ha desarrollado con éxito proyectos similares para otras organizaciones. El proveedor ha tenido experiencias exitosas con otros proveedores empleados previamente por la organización contratante.

Tabla 5.11. Formalidad de los procesos vs. experiencia previa con el proveedor. Basado en [SVE01].

<b>Formalidad del Proceso</b>	<b>Experiencia Previa con el Proveedor</b>
Robusto	No se conoce bien el proveedor o este ha tenido malos resultados en otras organizaciones. El proveedor ha tenido malas experiencias con otros proveedores previamente empleados en la organización contratante.

Tabla 5.11. Formalidad de los procesos vs. experiencia previa con el proveedor. Basado en [SVE01].

#### 5.1.2.1.2.3.4. *Dispersión Geográfica*

La ubicación geográfica de las partes involucradas en el proceso de adquisición es importante para cualquier tipo de proyecto. Cuando la organización contratante y el proveedor se encuentran sitios geográficos distantes, las labores de coordinación de las tareas del proyecto se vuelven más complicadas. En estos casos, es muy importante elaborar un plan de comunicaciones que desarrolle una estrategia apropiada para mitigar el impacto de la dispersión geográfica. La tabla 5.12 describe el nivel de formalidad requerido en el proceso de adquisición de acuerdo a la distancia geográfica existente con los proveedores:

<b>Formalidad del Proceso</b>	<b>Distancia Geográfica con los Proveedores</b>
Liviano	La organización contratante, el proveedor y otras partes involucradas están localizadas en un mismo sitio o es posible visitarse mutuamente en los términos de un día.
Controlado	La organización contratante, el proveedor y otras partes involucradas no se encuentran localizadas en el mismo sitio, pero los equipos asignados por estas si lo están. La comunicación sobre la base de un día no es posible.

Tabla 5.12. Formalidad de los procesos vs. distancia geográfica con los proveedores. Basado en [SVE01].

Formalidad del Proceso	Distancia Geográfica con los Proveedores
Robusto	La organización contratante, el proveedor y otras partes involucradas tienen sus respectivos equipos dispersos en distintas ubicaciones geográficas. La comunicación personal sobre la base de un día no es posible.

Tabla 5.12. Formalidad de los procesos vs. distancia geográfica con los proveedores. Basado en [SVE01].

### 5.1.2.2. Conformar Grupos para todos los Procesos de Adquisición

Después de haber adaptado los procesos de adquisición a las necesidades propias del proyecto, es necesario conformar los grupos y definir los roles que participarán activamente en el mismo. En este sentido, debe establecerse:

- Los roles requeridos en el proyecto
- Las habilidades necesarias para desempeñar dichos roles
- Conformación de los grupos que trabajarán en los distintos procesos
- Asignación de responsabilidades, autonomía y autoridad

Se debe tener en consideración, que las competencias de los participantes del proyecto de adquisición son un factor crítico de éxito y, en la medida de lo posible, se debe emplear el personal mejor capacitado para desempeñar las tareas definidas.

### 5.1.2.3. Desarrollar el Plan de Adquisición

El plan de adquisición tiene como propósito describir detalladamente como el proyecto de adquisición será manejado. Para dicho propósito, los siguientes elementos deben ser documentados:

- Definición del problema a ser resuelto, procedencia y sus antecedentes.

- Establecer los objetivos del proyecto.
- Definir la terminología que será empleada en proyecto de adquisición.
- Definir las restricciones presupuestales existentes.
- Describir las herramientas, técnicas y métodos que serán utilizados.
- Describir cualquier estándar, regulación, prácticas y convenciones a ser seguidas.
- Definir las estrategias para la resolución de problemas.
- Establecer los requisitos de documentación necesarios.
- Especificar los procedimientos de gestión de la configuración y del cambio.
- Describir las relaciones y contratos existentes con otras organizaciones (como contratos de soporte) y especificar como se encuentran divididas las responsabilidades entre ellas.
- Especificar los canales de comunicación y la jerarquía de escalado de reportes.
- Proyectar un presupuesto maestro para los diferentes procesos de adquisición.
- Esbozar un plan maestro donde se describa el ciclo de vida de la adquisición y los hitos del proyecto.
- Los procesos de adquisición a ser empleados y el nivel de personalización definido para los mismos.
- Planificar en el tiempo las actividades y presupuestarlas económicamente.
- Determinar como el equipo del proyecto se encuentra organizado (es documentar lo establecido en la actividad anterior)
- Especificar los procedimientos de confidencialidad, de acceso a la información y las restricciones de privacidad.
- Especificar los procedimientos para reportar el estado del proyecto.
- Describir como las mediciones serán obtenidas, analizadas y usadas.
- Determinar los recursos asignados al proyecto. Estos recursos pueden ser equipos de cómputo, herramientas software, documentos, servidores y similares.

#### 5.1.2.4. Revisar el Plan de Adquisición y Aprobar

El plan de adquisición debe ser revisado, validado y aprobado por los participantes del proyecto, es decir, por la organización contratante (en cabeza de los patrocinadores), por el proveedor, y por cualquier otra parte que participe en el proyecto y que haya adquirido algún tipo de compromiso con el mismo. Es preciso anotar que un plan de adquisición, al igual que un plan general de proyecto, traduce o materializa sus especificaciones en una serie de actividades que son organizadas en el tiempo de manera lógica, es decir, un cronograma. La tabla 5.13 describe un cronograma que propone las actividades y los hitos que como mínimo un plan de adquisición debería contener.

<b>Negociaciones Contractuales</b>	
Revisión de los temas relativos a los derechos de autor y la propiedad intelectual. En el proceso de Transición a Operaciones y Soporte (tabla 5.21) se ofrece una lista de chequeo que ayuda a clarificar estos temas.	<input checked="" type="checkbox"/>
Firmar el contrato ( <i>hito</i> ).	<input checked="" type="checkbox"/>
Fechas en las cuales el cliente suministrará al proveedor los recursos contractualmente requeridos (equipos, herramientas, servicios, espacios de trabajo, etc.) ( <i>hitos</i> ).	<input checked="" type="checkbox"/>
<b>Requisitos</b>	
Revisión de los requisitos.	<input checked="" type="checkbox"/>
Firmar los requisitos ( <i>hito</i> ).	<input checked="" type="checkbox"/>
Desarrollo rápido de prototipos.	<input checked="" type="checkbox"/>
<b>Estimaciones del Proyecto</b>	
Estimaciones de tamaño, tiempos y costos realizadas por el cliente.	<input checked="" type="checkbox"/>
Estimaciones de tamaño, tiempos y costos realizadas por el proveedor.	<input checked="" type="checkbox"/>
Resolver diferencias y llegar a acuerdos.	<input checked="" type="checkbox"/>

Tabla 5.13. Lista de chequeo para el cronograma del proyecto de adquisición. Basado en [SAA98].

<b>Controles de Gestión</b>	
Revisiones de administración de riesgos.	<input checked="" type="checkbox"/>
Revisiones del proyecto.	<input checked="" type="checkbox"/>
Inspecciones.	<input checked="" type="checkbox"/>
Revisiones de documentos.	<input checked="" type="checkbox"/>
Aprobación de documentos ( <i>hitos</i> ).	<input checked="" type="checkbox"/>
<b>Pruebas de Aceptación</b>	
Planeación detallada de las pruebas de aceptación.	<input checked="" type="checkbox"/>
Desarrollo de las pruebas de aceptación.	<input checked="" type="checkbox"/>
Análisis de los resultados de las pruebas.	<input checked="" type="checkbox"/>
Aceptación del producto software ( <i>hito</i> ).	<input checked="" type="checkbox"/>
<b>Capacitación</b>	
Preparación y planeación de las capacitaciones.	<input checked="" type="checkbox"/>
Desarrollo de las capacitaciones.	<input checked="" type="checkbox"/>
<b>Soporte</b>	
Desarrollo y/o ajuste de documentación técnica para el soporte (p.e., manuales de configuración, procedimientos para la resolución de problemas, bases de conocimiento, listas de chequeo, etc.).	<input checked="" type="checkbox"/>
Transición del producto a operaciones y mantenimiento ( <i>hito</i> ).	<input checked="" type="checkbox"/>

Tabla 5.13. Lista de chequeo para el cronograma del proyecto de adquisición. Basado en [SAA98].

## 5.2. Desarrollo de Requisitos

El propósito del proceso de Desarrollo de Requisitos es establecer, clarificar y definir el conjunto de requisitos que el producto software debe satisfacer, y que estos puedan ser entendidos tanto por la organización contratante, como por el proveedor y los participantes del proyecto.

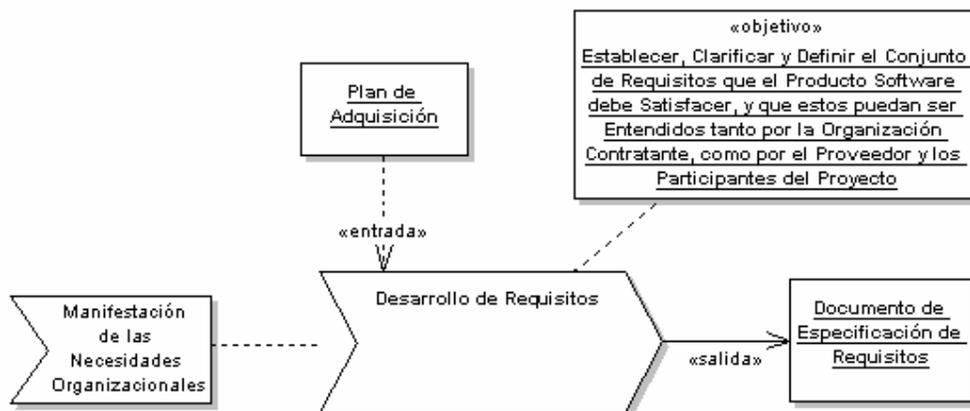


Figura 5.3. Diagrama de proceso de negocio del proceso de Desarrollo de requisitos.

La figura 5.3 muestra los elementos mas importantes del proceso de desarrollo de requisitos. El proceso inicia cuando el cliente (organización contratante) detecta que existen oportunidades de mejora, problemas o necesidades organizacionales que requieren ser resueltos de manera imperativa. Este proceso toma como insumo principal el plan de adquisición, del cual extrae la problemática global, las necesidades organizacionales, y las principales restricciones de recursos, tiempo y costos. Después de haber ejecutado el proceso, se obtiene como salida una especificación de requisitos que documenta de manera clara y concisa los requerimientos que el producto software debe reunir.

Para efectos del desarrollo delgado de software, el Desarrollo de Requisitos define dos contextos. El primero de ellos tiene que ver con la identificación, agrupación y coordinación de requisitos operacionales o, más precisamente, los requisitos cliente. Los requisitos cliente expresan los intereses, necesidades y expectativas de clientes y usuarios y, en este sentido, justifican la realización del proyecto de desarrollo. Adicionalmente, este tipo de requisitos son extremadamente útiles para definir y delimitar el alcance del proyecto y son el insumo principal para la elaboración de los requisitos del producto. El segundo contexto tiene que ver precisamente con la elaboración de los requisitos producto. Los requisitos producto son la extensión lógica de los requisitos cliente ya que definen las características que el producto software

debe poseer para satisfacer las necesidades de los mismos. Conjuntamente, los requisitos producto pueden definir restricciones de diseño, arquitecturas o patrones de desarrollo a obedecer, funcionalidades requeridas, restricciones de seguridad o cualquier otra característica que se deba incluir en el producto.

### **5.2.1. La Asignación de Responsabilidades en el Desarrollo de Requisitos**

Antes que nada, es importante tener claro que los requisitos pueden dividirse, clasificarse y agruparse de muchas maneras. Estas divisiones dependen de la situación particular de cada proyecto, pero en general puede decirse que existen requisitos de alto nivel y requisitos de bajo nivel. Los requisitos de alto nivel tienen que ver con los requisitos cliente y los de bajo nivel frecuentemente se asocian a los requisitos producto. Adicionalmente, puede reconocerse que entre los requisitos de alto nivel y los de bajo nivel pueden existir múltiples niveles y que estos niveles definen un conjunto de actividades a realizar y una serie de productos de trabajo (entregables). El paso, flujo de trabajo o recorrido por cada uno de estos niveles es precisamente el proceso de desarrollo de requisitos en el cual, bajo el contexto de la adquisición, se deben establecer, dividir y asignar las responsabilidades del proveedor y las responsabilidades de la organización contratante.

Respecto al proveedor, se puede generalizar la conclusión que este es responsable del desarrollo de los requisitos de bajo nivel, es decir, debe realizar las especificaciones del producto software. La organización contratante es responsable del desarrollo de los requisitos de alto nivel, lo cual sobrentiende un amplio conocimiento acerca de los intereses, necesidades y expectativas propias. Adicionalmente, se debe encargar de definir los distintos niveles a través de los cuales se desarrollarán los requisitos y establecer cuales de ellos estarán bajo su control.

Es importante también tener en cuenta que a pesar de la división de responsabilidades en el proceso de desarrollo de requisitos, la organización

contratante no se debe desentender de las actividades delegadas al proveedor. Lo anterior constituye un factor crítico de éxito y, por tanto, la organización contratante debe monitorear constantemente las especificaciones de requisitos realizadas en los niveles más bajos. Finalmente, es importante entender que el proceso de desarrollo de requisitos debe ser un proceso iterativo, ya que es muy pretencioso esperar que las especificaciones levantadas en etapas tempranas del proyecto expresen con certeza los requisitos cliente y los requisitos producto; los requisitos deben ser constantemente revisados y refinados a través del ciclo de vida del proyecto para asegurar que aglutinan tanto las necesidades de clientes, usuarios y negocio como las cualidades que el producto software debe implementar para satisfacerlas.

#### 5.2.1.1. Actividades del Proceso de Desarrollo de Requisitos

Objetivo Específico (SG)	Actividades / Tareas
Desarrollar los Requisitos Cliente	<ul style="list-style-type: none"> <li>• Entender las Necesidades de Clientes y Usuarios</li> <li>• Analizar el Problema</li> <li>• Definir el Sistema</li> <li>• Administrar Cambios en los Requisitos</li> </ul>
Desarrollar los Requisitos Producto	<ul style="list-style-type: none"> <li>• Refinar la Definición del Sistema</li> <li>• Desarrollar el Plan de Desarrollo de SW</li> <li>• Analizar el Comportamiento</li> </ul>
Analizar y Validar Requisitos	<ul style="list-style-type: none"> <li>• Reiterar / Analizar / Validar las Actividades de Requisitos</li> <li>• Candidatizar una Arquitectura</li> <li>• Crear y Evaluar Prototipos</li> </ul>

Tabla 5.14. Objetivos específicos del Desarrollo de Requisitos vs. Tareas a llevar a cabo. Basado en [GAB01].

La tabla 5.14 describe las actividades o tareas a realizar para alcanzar los objetivos específicos de cualquier proceso de desarrollo de requisitos. Bajo el contexto del desarrollo delegado de software y tomando como base los argumentos expuestos en la sección anterior, es posible reclasificar los objetivos y las actividades mostradas en la tabla con base en la división de responsabilidades entre la organización contratante y el proveedor. En este sentido, dicha tabla se redefiniría como sigue:

<b>Responsable</b>	<b>Objetivo Específico (SG)</b>	<b>Actividades / Tareas</b>
<b>Organización Contratante</b>	Desarrollar los Requisitos Cliente	<ul style="list-style-type: none"> <li>• Entender las Necesidades de Clientes y Usuarios</li> <li>• Analizar el Problema</li> <li>• Definir el Sistema</li> <li>• Administrar Cambios en los Requisitos</li> </ul>
<b>Proveedor de Software (monitoreo / revisión por parte de la Organización Contratante)</b>	Desarrollar los Requisitos Producto	<ul style="list-style-type: none"> <li>• Refinar la Definición del Sistema</li> <li>• Desarrollar el Plan de Desarrollo de SW</li> <li>• Analizar el Comportamiento</li> </ul>
	Analizar y Validar Requisitos	<ul style="list-style-type: none"> <li>• Reiterar / Analizar / Validar las Actividades de Requisitos</li> <li>• Candidatar una Arquitectura</li> <li>• Crear y Evaluar Prototipos</li> </ul>

Tabla 5.15. Objetivos, actividades y división de responsabilidades entre la organización contratante y el proveedor.

La tabla anterior expresa claramente la división de responsabilidades entre la organización contratante y el proveedor de software. Como se mencionó anteriormente, la organización contratante no puede desentenderse de las actividades delegadas al proveedor; por el contrario, debe desarrollar mecanismos que le permitan ejercer un control efectivo y eficiente sobre las mismas.

Respecto a las actividades y tareas a desarrollar, se toma como base la propuesta desarrollada por [RUP03] en cuanto a la gestión de requisitos se refiere, y se relacionan dichas actividades con los objetivos propuestos por el [SEI02] para la gestión de los mismos. Dicha relación se fundamenta en el análisis realizado por el propio SEI, el cual se encuentra documentado en [GAB01].

Por último y como se planteó anteriormente, este trabajo observa el proceso de desarrollo de software del lado las empresas usuarias y, en este sentido, se exponen las actividades a desarrollar por las mismas en cuanto al proceso de desarrollo de requisitos.

## **5.2.2. Objetivos**

- Entender la problemática organizacional
- Elicitar, analizar y validar las necesidades organizacionales
- Recopilar y coordinar los requerimientos de clientes y usuarios
- Establecer los requisitos del cliente
- Establecer los requisitos iniciales del producto

## **5.2.3. Actividades a Desarrollar**

- Analizar el Problema
- Entender las Necesidades de Clientes y Usuarios
- Desarrollar Especificaciones Suplementarias
- Definir el Sistema
- Desarrollo de la Especificación de Requisitos

### **5.2.3.1. Analizar el Problema**

El principal objetivo de Analizar el Problema, es definir de forma acordada el problema a ser resuelto. Esta actividad supone la identificación de participantes en el proyecto y la definición del alcance y las restricciones a ser impuestas en el sistema.

El proceso de analizar el problema involucra las siguientes tareas:

#### **5.2.3.1.3. Capturar un Vocabulario Común**

Esta tarea describe como definir el conjunto de términos comunes que necesitan ser referenciados consistentemente en el proyecto.

#### 5.2.3.1.3.1. Encontrar Términos Comunes

Para la evaluación de los requisitos y para el resto del proyecto se debe definir un conjunto de términos comunes acordes con la definición del problema y estos se deben recopilar en un glosario y ponerse a disposición de todos los implicados en el proyecto de tal modo que se mitigue el impacto generado por malentendidos o confusiones.

Con el fin de identificar términos relevantes se puede recurrir a los requerimientos de los participantes del proyecto o al conocimiento general de lo que para estos debería ser el sistema. Básicamente, estos términos se presentan de las siguientes formas:

- Como conceptos de negocio que son referenciados constantemente en las labores cotidianas de la organización.
- Como conceptos que hacen referencia al entorno operativo donde se espera que el sistema actúe.
- Como objetos del mundo real de los que el sistema necesita ser consciente.

#### 5.2.3.1.4. Buscar Actores y Casos de Uso

Esta tarea tiene como propósito la identificación de los actores y los casos de uso que soportarán la implementación de los requisitos de clientes y usuarios. Identificando los actores y los casos de uso se define explícitamente el alcance del sistema.

##### 5.2.3.1.4.1. Encontrar Actores

Encontrar los actores es uno de los primeros pasos para definir el uso del sistema. Cada tipo de fenómeno externo con el cual el sistema debe interactuar es representado por un actor. Para encontrar los actores del sistema, es importante dar solución a las siguientes preguntas:

- ¿Qué usuarios requieren ayuda del sistema para realizar sus labores?
- ¿Qué usuarios se requieren para ejecutar las funciones principales del sistema?
- ¿Qué usuarios se requieren para ejecutar funciones secundarias, tales como mantenimiento del sistema o administración del mismo?
- ¿El sistema interactuará con algún sistema hardware o software externo?

Cualquier grupo individual o fenómeno externo que encaje en una o más de estas categorías es un candidato para ser un actor del sistema.

Para determinar si se han identificado completamente los actores (humanos) adecuados, se recomienda nombrar dos o tres personas que se puedan desempeñar como actores, y después se debe analizar si este grupo de actores es suficiente para ejecutar las funcionalidades del sistema que darán solución a los requisitos establecidos.

Puede ser difícil en principio encontrar los actores mas adecuados, y lo más probable es que no se encuentren inmediatamente dado que esta labor es altamente dependiente de los casos de uso que se definan. Trabajar con casos de uso es la estrategia mas recomendada para lograr un conocimiento profundo del entorno del sistema y como se interactúa con el mismo.

#### 5.2.3.1.4.2. Nombrar y Describir Brevemente los Actores Identificados

El nombre que se le de al actor debe claramente denotar el rol del mismo. Además, al realizar este nombramiento, se debe propender por minimizar el riesgo que en una etapa futura del proyecto se confunda el nombre de un actor con el de otro.

Respecto a la definición del actor, se debe escribir una descripción breve que incluya el área de responsabilidad del mismo y definir de manera concreta las funcionalidades (necesidades) que este explotará del sistema. Dado que los actores representan elementos externos al sistema, no es necesario realizar una descripción detallada de los mismos.

#### 5.2.3.1.4.3. Encontrar Casos de Uso

Cuando el primer borrador de la definición de actores se encuentre completo, se debe proseguir con la búsqueda e identificación de los casos de uso del sistema. Los primeros casos de uso identificados serán algo incipientes, y sin lugar a dudas deberán ser redefinidos o cambiados varias veces hasta que alcancen un buen grado de estabilidad. Si la visión del sistema o los requisitos son deficientes, o si el análisis del mismo se realiza con ligereza, la funcionalidad del sistema será poco clara. Por lo tanto, se debe asegurar que los casos de uso identificados son los necesarios para expresar la funcionalidad requerida y que estos se encuentren debidamente definidos.

La mejor manera de encontrar casos de uso es considerar lo que cada usuario requiere del sistema. Se debe tener en cuenta que el sistema existe solo por sus usuarios y debe, por consiguiente, estar basado en sus necesidades. Para identificar las necesidades de los actores es absolutamente necesario referirse a los requisitos funcionales previamente definidos. Para cada actor, sea este humano u otro sistema, se deben tener en cuenta los siguientes aspectos:

- ¿Cuáles son las tareas primarias que el actor quiere que el sistema realice?
- ¿El actor creará, guardará, cambiará, removerá, o consultará información contenida en el sistema?
- ¿Necesitará el actor informar al sistema de cambios externos repentinos?
- ¿Necesitará el actor ser informado acerca de ciertas ocurrencias o eventos en el sistema?

- ¿Realizará el actor el inicio o apagado del sistema?

Las respuestas a estas preguntas representan el flujo de eventos que definen los casos de uso candidatos. No todos los casos de uso constituyen casos de uso separados; algunos podrían ser modelados como variantes del mismo caso de uso. No siempre es fácil discernir entre cual caso de uso es una variante de algún otro caso de uso o si estos representan casos de uso completamente distintos e independientes. Sin embargo, ese discernimiento se hará más claro a medida que se describa el flujo de eventos en forma detallada para cada caso de uso.

Respecto al modelo de casos de uso, la funcionalidad de un sistema podría ser representada a través de muchos posibles modelos de casos de uso. La mejor manera de encontrar el modelo óptimo es desarrollar dos o tres modelos y escoger el que mejor represente la funcionalidad requerida y desarrollarlo completamente. Desarrollar algunos modelos alternativos, también es recomendable para entender mejor el sistema.

#### 5.2.3.1.4.4. Nombrar y Describir Brevemente los Casos de Uso Identificados

Cada caso de uso debe definir un nombre que indique lo que se logra cuando un actor interactúa con él. El nombre puede contener varias palabras para que sea entendible y se debe propender porque este no sea utilizado por varios casos de uso, de modo que se eviten posibles confusiones. Adicionalmente, se debe definir cada caso de uso con una breve descripción del mismo. Mientras se escribe esta descripción es bastante útil referirse al glosario y, si es necesario, se deben definir nuevos conceptos. La tabla 5.16 elabora una lista de chequeo que especifica los elementos que como mínimo se deben incluir en una especificación de un caso de uso:

Para cada caso de uso se debe incluir:	
¿Quién es el responsable de la especificación del caso de uso?	<input checked="" type="checkbox"/>
¿Quién es la fuente de suministro de la información?	<input checked="" type="checkbox"/>
Describir las dependencias que el casos de uso tiene con otros casos de uso y con los requerimientos de usuario definidos.	<input checked="" type="checkbox"/>
Se debe realizar una corta descripción del caso de uso que indique su propósito o finalidad.	<input checked="" type="checkbox"/>
Se deben especificar todas aquellas condiciones que deben satisfacerse para que la ejecución del caso de uso sea exitosa (precondiciones).	<input checked="" type="checkbox"/>
Se desarrolla una secuencia de acciones que guían paso a paso la ejecución del caso de uso (flujo de eventos). (Nota. Aunque esta especificación hace mas parte de un fase inicial de diseño, se debe velar en todo momento por su cumplimiento).	<input checked="" type="checkbox"/>
Se especifican los flujos de eventos encargados de manejar cualquier comportamiento excepcional del caso de uso (Nota. Aplican las mismas consideraciones de la nota anterior).	<input checked="" type="checkbox"/>
Se describen las condiciones que deben satisfacerse después de la ejecución del caso de uso para que se garantice la estabilidad el producto (poscondiciones).	<input checked="" type="checkbox"/>
Se asocia al caso de uso un atributo que indica su importancia frente al proceso de negocio que apoya. Este atributo sirve como criterio de priorización.	<input checked="" type="checkbox"/>
Se asocia al caso de uso un atributo que indica la urgencia con la que debe ser tratado, dada su importancia y la imperatividad de las necesidades organizacionales. Este atributo, al igual que en el caso anterior, sirve como criterio de priorización.	<input checked="" type="checkbox"/>
El caso de uso define un estado el cual determina si se encuentra en construcción, si esta pendiente por alguna validación o si ha sido cancelado.	<input checked="" type="checkbox"/>
Se especifica la estabilidad del caso de uso, es decir, se resuelve la pregunta ¿qué probabilidad existe que la definición del caso de uso pueda ser modificada? Dado que exista esta probabilidad, ¿en que plazo se podría presentar la modificación? La anterior situación define un riesgo que debe ser tomado en cuenta ya que, en caso contrario, se pueden perder los esfuerzos realizados en el diseño y la codificación del caso de uso.	<input checked="" type="checkbox"/>

Tabla 5.16. Lista de chequeo para especificaciones de casos de uso<sup>24</sup>.

#### 5.2.3.1.4.5. Esbozar el Flujo de Eventos

En este punto, se debe elaborar un borrador del flujo de eventos del caso de uso. Con este fin, se recomienda describir el flujo de eventos del caso de uso en forma de breves instantes de rendimiento, teniendo cuidado de no entrar en detalles dada la marcada insipiencia de los mismos. Los analistas que en una fase más madura del proyecto especificarán de forma completa el caso de uso, necesitarán hacer referencia a la descripción elaborada. En este sentido, se debe empezar por esbozar el flujo básico de eventos, y una vez finalizada esta tarea se deben identificar y esbozar los flujos alternativos.

<sup>24</sup> Basado en: [DUB00], p. 36.

#### 5.2.3.1.4.6. Recopilar Requisitos Adicionales

A través de la realización de las tareas relativas a la identificación y descripción de los casos de uso, es muy probable que salga a flote el hecho que algunos de los requisitos del sistema no puedan ser ligados a casos de uso específicos. Esto se debe a la naturaleza de estos requisitos que, probablemente, sean no funcionales. En este caso, se debe recopilar esa información y plasmarla en las especificaciones suplementarias.

#### 5.2.3.1.4.7. Describir las Interacciones entre Actores y Casos de Uso

Dada la importancia de mostrar como los actores se relacionan con los casos de uso se debe, para cada caso de uso en particular, establecer los actores que interactuarán con el. Para dicho propósito, es necesario definir una asociación de comunicación entre el actor y el caso de uso; dicha asociación debe ser navegable en la misma dirección que se establece la comunicación entre estos artefactos<sup>25</sup>. Adicionalmente, es necesario describir brevemente cada asociación que se defina, es decir, se le debe dar un nombre que describa claramente la naturaleza de la asociación.

#### 5.2.3.1.4.8. Empaquetar Casos de Uso y Actores

Si el número de actores o casos de uso se vuelve muy grande, es muy recomendable agruparlos en “paquetes de casos de uso”, lo cual simplifica el mantenimiento del modelo de casos de uso. Como beneficios adicionales, se tiene que el empaquetamiento hace que el modelo de casos de uso sea fácil de entender y simplifica la asignación de responsabilidades dejando que los desarrolladores sean garantes de paquetes de casos de uso o de actores en particular.

---

<sup>25</sup> Bajo el contexto de este trabajo, el término artefacto tiene la misma connotación que los términos producto o producto de trabajo.

Algunos de los criterios que pueden utilizarse para agrupar casos de uso pueden ser:

- Casos de uso que interactúan con el mismo actor.
- Casos de uso que comparten relaciones de inclusión, extensión o generalización.
- Casos de uso que expresan funcionalidades relacionadas o que manipulan objetos de negocio fuertemente relacionados.

#### 5.2.3.1.4.9. Presentar el Modelo de Casos de Uso en Diagramas

Es recomendable ilustrar gráficamente las relaciones existentes entre casos de uso y entre casos de uso y actores. Esta ilustración se elabora mediante diagramas del modelo de casos de uso, los cuales pueden contener algunos de los siguientes elementos:

- Actores pertenecientes al mismo paquete de casos de uso.
- Un actor y todos los casos de uso con los que interactúa.
- Casos de uso que trabajen objetos de negocio fuertemente relacionados (subsistemas).
- Casos de uso utilizados por un mismo grupo de actores.
- Casos de uso que son frecuentemente ejecutados en cierta secuencia de eventos.
- Casos de uso que pertenecen al mismo paquete de casos de uso.
- Los casos de uso más importantes. Un diagrama de este estilo puede funcionar a manera de sumario del modelo, y es muy probable que sea incluido en la vista de casos de uso.

Por último, es necesario que cada diagrama sea incluido en el paquete apropiado del modelo de casos de uso.

#### 5.2.3.1.5. Explorar el Modelo de Casos de Uso

Con el fin de asegurar la completitud del modelo de casos de uso, esta tarea propone la elaboración de un sondeo, procedimientos de prueba o examen del mencionado modelo. En este sondeo, se recomienda incluir los siguientes elementos:

- Secuencias típicas en las cuales los casos de uso son empleados por los usuarios.
- Funcionalidad que no se maneje o que no este contenida explícitamente en el modelo de casos de uso.

Finalmente, con los resultados obtenidos, se puede valorar la completitud del modelo de casos de uso y, si es necesario, refinarlo adecuadamente.

#### 5.2.3.2. Entender las Necesidades de Clientes y Usuarios

Por medio de la obtención de información acerca del producto deseado, esta actividad busca entender las necesidades de clientes y usuarios.

Dependiendo si el producto a desarrollar corresponde a un ajuste o mejora de un sistema existente, o si concierne a un sistema completamente nuevo, se deben llevar a cabo o refinar los productos desarrollados en las siguientes tareas:

- Capturar un Vocabulario Común
- Desarrollar una Visión
- Buscar Actores y Casos de Uso

Además de lo anterior, Entender las Necesidades del Cliente define las siguientes actividades:

#### 5.2.3.2.1. Elicitar los Requisitos de Clientes y Usuarios

Esta tarea describe como gestionar los requisitos de los participantes del proyecto acerca de las funcionalidades que desean que el sistema provea.

##### 5.2.3.2.1.1. Determinar las Fuentes de los Requisitos

Para un sistema existente, el primer grupo de entradas a esta tarea será un conjunto de requisitos de mejora que han sido recolectados y pospuestos a través del ciclo de vida del producto.

Por otro lado, es importante identificar y definir un subconjunto de participantes que servirán como “fuente de requisitos”, para lo cual se debe considerar sus conocimientos, sus habilidades de comunicación, su disponibilidad y su experiencia. Con este grupo de participantes se conformará una especie de equipo extendido del proyecto y, en este sentido, debe considerarse que el tamaño del grupo tiene incidencia directa en la manejabilidad y eficiencia del mismo: se recomienda que este grupo no pase de 5 personas. Adicionalmente y a saber, los integrantes de este equipo no requerirán de total disponibilidad para adelantar las tareas del proyecto ya que, ocasionalmente, deberán participar en talleres de requisitos y posteriormente en ágiles sesiones de revisión.

Finalmente, y como recomendación, es importante recurrir a las visitas en sitio para analizar como los usuarios están utilizando el sistema y de este modo identificar aspectos que sean susceptibles de mejora. También es muy recomendable recurrir a modelos de negocio y a modelos de proceso desarrollados en proyectos diferentes para identificar procesos de negocio claves que pueden ser apoyados por mejoras realizadas a algún sistema de información existente, o con el desarrollo de un nuevo sistema.

#### 5.2.3.2.1.2. Obtener Información

Esta tarea tiene como objetivo formular las preguntas que necesitan ser resueltas y obtener y documentar información relativa a las necesidades y requisitos de los participantes. A continuación se describen brevemente algunas de las técnicas empleadas para la elicitación de requisitos:

##### 5.2.3.2.1.2.1. *Guiones Gráficos*

Los guiones gráficos o más comúnmente conocidos como “storyboards”, definen una técnica que se puede adaptar a la elicitación de requisitos y que básicamente lo que pretende es describir a través de narraciones gráficas los distintos escenarios que el sistema deberá soportar [WEI98].

##### 5.2.3.2.1.2.2. *Entrevistas*

Las entrevistas son una importante técnica para elicitar información detallada de un individuo. Esta es comúnmente usada en la elicitación de requisitos para sistemas grandes como parte de alguna técnica de elicitación de alto nivel. Además, puede ser usada en proyectos pequeños como la única técnica de elicitación de requisitos<sup>26</sup>.

##### 5.2.3.2.1.2.3. *Cuestionarios*

Los cuestionarios son otra técnica de elicitación de requisitos ampliamente utilizada. A medida que se conducen las entrevistas puede encontrarse que existen grandes similitudes en las respuestas e impresiones proporcionadas por los participantes del proyecto. Si la información proporcionada empieza a ser muy repetitiva y si el número de participantes es elevado, es muy recomendable elaborar cuestionarios ya que de este modo se acude a la

---

<sup>26</sup> [RZF97], p. 50.

practicidad, se clarifican requisitos y se pueden obtener respuestas más certeras acerca de las necesidades de los participantes. Pese a esto, los cuestionarios no son una técnica infalible ya que se puede caer en el error de asumir la relevancia de las preguntas que se les hacen a los participantes, y como consecuencia obtener resultados carentes de importancia.

#### 5.2.3.2.1.3. Promover Talleres de Requisitos

Los talleres de requisitos se realizan con el propósito que el equipo de trabajo conozca a los participantes del proyecto, además de obtener una lista comprensiva de las expectativas, deseos y requisitos respecto a la solución informática que se pretende desarrollar. Adicionalmente, esta tarea debe ayudar a priorizar los requisitos levantados, tomando como base las necesidades más imperativas y el esquema de relevancia planteado por los participantes.

Por último y para propósitos de referencia, es de aclarar que los talleres de requisitos emplean técnicas de elicitación de requisitos para extraer los requisitos del sistema y hacer un filtro entre lo que se desea y lo que verdaderamente se necesita. En la literal “Obtener Información” se describieron brevemente algunas de las técnicas de elicitación más comúnmente utilizadas, pero se recomienda referirse a la literatura para obtener una descripción más amplia de estos conceptos.

#### **5.2.3.3. Desarrollar Especificaciones Suplementarias**

Esta tarea tiene como propósito capturar y especificar los requisitos del sistema que no pueden ser representados a través de casos de uso, en otras palabras, los requisitos no funcionales [RUP03].

#### 5.2.3.3.2. Capturar Requisitos no Funcionales

Los requisitos funcionales describen los comportamientos (funciones o servicios) del sistema que soportan los objetivos, tareas o actividades desempeñadas por los usuarios.

Mientras que muchos de los requisitos funcionales pueden ser documentados a través de casos de uso, existen otro tipo de requisitos que por su naturaleza no pueden ser asociados a casos de uso específicos. Estos requisitos se denominan requisitos no funcionales y representan características que el sistema debe poseer tales como velocidad, rendimiento, fácil mantenibilidad, fiabilidad, plataforma tecnológica y cualquier otra restricción del entorno o de la implementación que se necesite definir<sup>27</sup>. Los requisitos no funcionales son definidos y mantenidos en un documento de especificaciones suplementarias.

#### 5.2.3.3.3. Capturar los Atributos de Calidad del Sistema

Los atributos de calidad del sistema son propiedades o características del mismo que afectan el grado de satisfacción de los usuarios con la solución informática desarrollada. Además de la funcionalidad requerida en el sistema (la cual se encuentra representada por los casos de uso), según la norma ISO 9126 los atributos de calidad son los siguientes [ISO91]:

- Usabilidad: estética y consistencia de la interfaz de usuario.
- Confiabilidad: disponibilidad (la cantidad de tiempo disponible del sistema), certeza en los cálculos realizados por el sistema y su capacidad para el manejo de errores.
- Desempeño: tasa de transferencia, tiempos de respuesta, tiempo de recuperación, tiempo de inicialización, tiempo de apagado.

---

<sup>27</sup> [JBR00], p. 110.

- **Mantenibilidad:** facilidad para llevar a cabo pruebas, adaptabilidad, compatibilidad, configurabilidad, instalabilidad, escalabilidad, y localizabilidad.

Dependiendo del número de atributos de calidad que se definan para el sistema, es recomendable agruparlos en secciones de acuerdo a la tipología de los mismos.

#### 5.2.3.3.4. Capturar Restricciones

En esta tarea se deberá documentar cualquier restricción de diseño que se requiera en el sistema. En términos generales, una restricción es una limitación en el grado de libertad que se tiene para desarrollar la solución o alguna condición de diseño que esta deba satisfacer. Habitualmente, las restricciones se ajustan a alguna de las siguientes categorías:

- **Restricciones de diseño:** este tipo de restricción predefine o limita la arquitectura del sistema al cumplimiento de ciertas condiciones de diseño. Por ejemplo, que la persistencia de los datos manipulados por una aplicación se implemente en una base de datos relacional.
- **Restricciones de implementación:** este tipo de restricción predefine o limita la construcción del sistema al cumplimiento de ciertas condiciones de codificación. Por ejemplo, estándares de codificación, procesos, herramientas, lenguajes de programación, plataformas hardware, sistemas operativos, componentes software preexistentes, librerías de clases.
- **Restricciones de interfase:** especifica algún elemento externo con el cual el sistema debe interactuar, o condiciones en los formatos utilizados para llevar a cabo la interacción. Por ejemplo, interactuar con un sistema externo a través de colas de mensajes.

- Restricciones físicas: especifica condiciones físicas que el hardware que alojará el sistema debe satisfacer. Por ejemplo, la forma, el tamaño, o la marca del componente hardware.

Al igual que con los requisitos funcionales y no funcionales, dependiendo del número de restricciones definidas y de la tipología de las mismas, es importante agruparlas y documentarlas en diferentes secciones de las especificaciones suplementarias. Adicionalmente, es importante analizar las siguientes consideraciones:

- Si las restricciones definen la compra de componentes software de terceros (p.e., componentes COTS)<sup>28</sup>, se debe documentar en detalle el esquema de licenciamiento aplicable, las condiciones de uso, las especificaciones de compatibilidad/interoperabilidad asociadas y los estándares de interfase.
- Si las restricciones definen requisitos de interfase específicos, se debe documentar en secciones separadas los diferentes tipos de interfaces (hardware, software, usuario). Para cada interfase es preciso especificar los protocolos, los puertos, las direcciones lógicas y demás elementos necesarios para que el sistema pueda ser desarrollado y verificado a la luz de los requisitos establecidos. Específicamente:
  - Para interfaces de usuario, se deben describir las interfaces de usuario que el sistema debe implementar.
  - Para interfases hardware, se debe incluir la estructura lógica, las direcciones físicas y el comportamiento esperado de las interfases.
  - Para interfases software, se debe incluir una descripción de las interfases a otros componentes del sistema de

---

<sup>28</sup> Los componentes COTS se refieren a productos software genéricos que se encuentran disponibles en el mercado. Estos, generalmente, se encuentran bien definidos y documentados, y un amplio número de usuarios han probado su eficacia. Por lo general, el proveedor no adapta el producto a las necesidades particulares de los clientes y controla todo lo que tiene que ver con su mantenimiento. El costo de adquirir COTS es relativamente bajo, el tiempo de entrega es inmediato y el cliente no tiene control sobre la calidad del mismo [SVE01].

software. Estos componentes pueden ser comprados, reutilizados de otras aplicaciones o que están siendo desarrollados para subsistemas fuera del alcance del proyecto pero que con los cuales el sistema deberá interactuar.

#### 5.2.3.3.5. Capturar Requisitos de Cumplimiento

Por cumplimiento, se entiende el acatamiento de estándares (incluyendo estándares regulatorios, estándares de codificación o guías de estilo de interfaz de usuario), restricciones legales, garantías, derechos de autor, patentes y marcas y logotipos registrados.

Los requisitos de cumplimiento pueden definirse en términos de otros requisitos (funcionales, no funcionales y restricciones) y deben agruparse y documentarse como especificaciones suplementarias. Adicionalmente, y, en aras de simplificar su manejo, es recomendable hacer un compendio de los estándares y de las políticas que el sistema debe cumplir.

A continuación se describen brevemente algunos de los requisitos de cumplimiento que se deben tener en cuenta en cualquier proyecto de desarrollo de software:

#### 5.2.3.3.6. Requisitos de Licencias

Esta labor define las condiciones que el esquema de licenciamiento impone así como cualquier otra restricción de uso que deba ser exhibida por el sistema software.

#### 5.2.3.3.7. Derechos de Autor y Avisos Legales

Esta tarea implica la descripción detallada de avisos legales, garantías, derechos de autor y patentes, así como también de palabras, marcas y logotipos registrados. En el proceso de Transición a Operaciones y Soporte (tabla 5.16) se ofrece una lista de chequeo que ayuda a clarificar los temas concernientes a la propiedad intelectual.

#### 5.2.3.3.8. Estándares Aplicables

En esta etapa se debe describir (por referencia) cualquier estándar utilizable y las secciones específicas donde cualquiera de ellos aplique al sistema que esta siendo descrito. Estos estándares pueden referirse a estándares legales, regulatorios o de calidad; también pueden referirse a estándares de la industria para la usabilidad, interoperabilidad, internacionalización y conformidad con determinados sistemas operativos.

#### 5.2.3.3.9. Capturar Requisitos de Documentación

En este paso, se describen los requisitos de documentación que se consideren necesarios. Los requisitos de documentación pueden incluir el desarrollo de ayudas en línea, como también documentación de usuario final como los son las guías de instalación y configuración, manuales de usuario y material de entrenamiento.

En forma similar a los requisitos de cumplimiento, los requisitos de documentación conducen a la definición de otro tipo de requisitos, como lo son los requisitos funcionales (el sistema debe soportar el acceso funcional a la ayuda en línea...), y los requisitos de usabilidad: la facilidad en el acceso a la información sobre el uso del sistema hace que este sea más usable.

#### **5.2.3.4. Definir el Sistema**

Lo que se pretende en esta actividad es lograr una convergencia entre los participantes del proyecto en cuanto a lo que debe ser el alcance del sistema. En este sentido, se hace imprescindible proyectar la complejidad de los requisitos de alto nivel mediante una descripción general de los requisitos del producto, para lo cual la definición de estos últimos debe trabajar y aclarar los siguientes aspectos:

- Que los miembros del equipo del proyecto tengan una visión unificada de lo que debe ser el producto software.
- La realización de un análisis de alto nivel respecto a los requisitos elicitados.
- Captura de las características clave del producto software a través del refinamiento de la visión.
- Refinamiento del Modelo de Casos de Uso para que incluya todos los casos de uso relevantes para el producto software.
- Capturar en las Especificaciones Suplementarias los requisitos no funcionales.
- Estructurar de una manera mas adecuada los resultados de las actividades de elicitación.

La definición del producto software se concentra en la identificación y especificación de manera más detallada de actores y casos de uso, así como también de los requisitos no funcionales.

##### **5.2.3.4.1. Desarrollar la Visión Global del Producto**

Esta tarea describe como desarrollar una visión global del producto software, incluyendo la definición del problema a ser resuelto, los integrantes del equipo del proyecto, el alcance del sistema, sus características clave, y cualquier restricción que deba ser impuesta sobre el mismo.

#### 5.2.3.4.1.1. Llegar a un Acuerdo Sobre el Problema a Resolver

Una de las formas más simples para llegar a un acuerdo sobre la definición del problema, es escribirla e indagar si los participantes clave del proyecto concuerdan con dicha definición. En este sentido, es importante desarrollar con claridad los siguientes elementos:

Se debe formular al equipo del proyecto la pregunta: ¿cuál es el problema?

- Es un error muy común adelantarse a pensar en una solución en vez de primero invertir el tiempo que sea necesario para entender el problema. Es indispensable escribir el problema e intentar colocar de acuerdo a los participantes claves en la definición levantada, estos son, clientes, usuarios y analistas.

Dada la importancia y la complejidad que implica identificar cual es verdaderamente el problema a ser resuelto, es muy recomendable insistir en preguntarle al grupo: ¿cuál es realmente el problema?

- Se debe identificar y analizar la fuente del problema y establecer sus causas principales. Es muy común confundir el verdadero problema con una serie de incidentes relacionados que lo que finalmente indican es la existencia de una situación anómala de fondo. Visto de otro modo, es frecuente que el verdadero problema se oculte detrás de lo que se percibe como un problema, es decir, el "problema detrás del problema".

No se debe aceptar la primera definición del problema; es necesario insistir en preguntar el porque de las cosas y explorar a fondo la naturaleza de la situación anómala. Es frecuente que los participantes del proyecto se encuentren tan concentrados en la solución, que pierden la visión global del problema. En estos casos, es bueno explorar los beneficios de la solución

propuesta para luego identificar los problemas que se resuelven con dichos beneficios; luego se debe analizar si los problemas encontrados se corresponden con la visión global del equipo en cuanto a lo que el problema verdaderamente representa y así lograr un común acuerdo.

#### 5.2.3.4.1.2. Identificar y Formalizar los Usuarios del Producto

Dependiendo de la experiencia del equipo desarrollador, identificar los usuarios del producto puede ser o no, una tarea trivial. Muchas veces esta labor solamente implica entrevistar a las personas con la suficiente autonomía para tomar decisiones, a los usuarios potenciales y a otros grupos interesados. Para facilitar la identificación de los usuarios, es necesario profundizar en los siguientes aspectos:

- ¿Quiénes son los usuarios potenciales del producto software?
- ¿Quién es el patrocinador o dependencia patrocinadora del producto software (la dependencia organizacional que asumirá los costos)?
- ¿Quiénes se verán afectados por las salidas que el producto producirá?
- ¿Quién evaluará y aprobará el producto software cuando sea entregado y liberado?
- ¿Existen otros usuarios internos o externos cuyas necesidades deban ser atendidas?
- ¿Quién adelantará las labores de mantenimiento sobre el sistema?
- ¿Realmente se han identificado todos los usuarios y participantes relevantes?

Finalmente, es necesario empezar a desarrollar perfiles de los usuarios potenciales del sistema. La información inicial de los usuarios clave y su entorno debe ser plasmada en el documento de visión.

#### 5.2.3.4.1.3. Definir los Límites del Sistema

El alcance del sistema define la línea divisoria entre la solución y el mundo real que rodea la solución; dicho de otro modo, el alcance describe una envoltura en la cual se encuentra contenido el sistema solución. La información es representada en la forma de datos de entrada y salida, y es transmitida desde y hacia el sistema por usuarios que residen por fuera de los límites de la aplicación. Cualquier interacción con el sistema ocurre por medio de interfaces entre este y el mundo exterior.

En muchos casos, los límites del sistema son fácilmente identificables. Las interfaces entre el usuario y la aplicación consisten en formularios o cuadros de diálogo que el usuario utiliza para ingresar información al sistema, y en cualquier reporte o ruta de comunicación que la aplicación utilice para documentar o transmitir la información resultante.

#### 5.2.3.4.1.4. Identificar las Restricciones a ser Impuestas en el Sistema

Existen varias fuentes de restricciones a ser consideradas. A continuación se esbozan algunas de estas fuentes y aspectos a considerar:

- Políticos: ¿Existen limitaciones a nivel de políticas internas o externas a la organización que afecten la solución potencial? ¿Existen relaciones inter departamentales? ¿El proyecto es transversal?
- Económicos: ¿Qué limitaciones financieras o presupuestales son aplicables? ¿Existen consideraciones de precios de productos? ¿Existen consideraciones de licenciamiento?
- Ambientales: ¿Existen limitaciones en la forma de regulaciones legales o ambientales? ¿Existen estándares que deban ser seguidos?
- Técnicos: ¿Se debe utilizar una tecnología en particular? ¿Se debe trabajar con plataformas o tecnologías existentes? ¿Existen limitaciones para trabajar con alguna tecnología emergente?

- Viabilidad: ¿Se ha predefinido algún cronograma? ¿Se cuenta únicamente con los recursos existentes? ¿Se tiene la posibilidad de tercerizar alguna labor? ¿Se cuenta con la posibilidad de incorporar más recursos humanos de forma temporal o permanente?
- De Sistema: ¿La solución a ser desarrollada se implementará o asociará con la funcionalidad de algún sistema existente? ¿Se debe mantener compatibilidad con las soluciones existentes? ¿Qué sistemas operativos y entornos deben ser soportados?

#### 5.2.3.4.1.5. Formular la Declaración del Problema

Con el propósito de diferenciar problemas de soluciones y preguntas de respuestas y así facilitar la declaración formal del problema, se propone diligenciar con todo el grupo de trabajo una ficha bibliográfica por cada problema que se haya identificado.

La ficha bibliográfica debe contener la descripción del problema, los participantes o usuarios que son afectados por el mismo, el impacto que este tiene a nivel de negocio u operativo, y la estrategia que probablemente daría solución exitosa a esta anomalía.

#### 5.2.3.4.1.6. Definir las Características del Sistema

En esta tarea, se debe definir una lista de las características que el sistema debe poseer tomando como base las declaraciones de los problemas desarrolladas en el paso anterior y las soluciones propuestas. Es necesario que las características identificadas sean descritas brevemente y que se les otorgue atributos que permitan identificar su estado y la prioridad con la cual serán trabajadas durante el proyecto.

### 5.2.3.5. Desarrollo de la Especificación de Requisitos

Después de haber recabado, analizado y filtrado toda la información obtenida de clientes y usuarios, y con un último análisis de los resultados obtenidos en cada una de las tareas desarrolladas, no queda mas que elaborar el documento de especificación de requisitos. La tabla 5.17 desarrolla una lista de chequeo que define los elementos esenciales que deben verse contenidos en dicho documento y brinda algunas claves que deben ser tenidas en cuenta:

<b>Generalidades</b>	
Establecer los objetivos del producto software.	<input checked="" type="checkbox"/>
¿Qué objetivos se podrían sacrificar en caso que la organización contratante se vea obligada a realizar concesiones al proveedor?	<input checked="" type="checkbox"/>
Descripción general del producto; antecedentes que promovieron la adquisición; tipo de producto esperado; dominio en el que será usado.	<input checked="" type="checkbox"/>
Describir los distintos tipos de usuario que el producto soportará. ¿Cómo usarán el producto? ¿Qué tiene que hacer el producto por ellos?	<input checked="" type="checkbox"/>
Definir la terminología usada en la especificación.	<input checked="" type="checkbox"/>
Establecer las suposiciones que hayan sido realizadas.	<input checked="" type="checkbox"/>
<b>Requisitos Funcionales</b>	
Que capacidades el sistema debe tener. Esto debe especificarse a nivel funcional y no prescribir la solución.	<input checked="" type="checkbox"/>
Cada funcionalidad requerida debe poder ser verificable mediante algún tipo de prueba y debe describirse con una sentencia de la forma "el sistema debe..." (p.e., el sistema deberá imprimir la factura después de haber ingresado el pedido).	<input checked="" type="checkbox"/>
Describir si la funcionalidad requerida es manual, automática o semiautomática (p.e., el sistema muestra un cuadro de dialogo al administrador para que este ingrese el mensaje que le debe aparecer a los usuarios cuando cierto evento ocurra).	<input checked="" type="checkbox"/>
Describir los requisitos de interfases de usuario. Esto se puede hacer de manera tan detallada como se requiera, pero es recomendable utilizar la técnica de los prototipos para agilizar esta labor.	<input checked="" type="checkbox"/>
Especificar los algoritmos, fórmulas o ecuaciones que se requieran.	<input checked="" type="checkbox"/>
Establecer los requisitos para el manejo de errores generados por el producto.	<input checked="" type="checkbox"/>
Establecer una prioridad a cada requisito.	<input checked="" type="checkbox"/>
Describir el riesgo asociado a cada requisito.	<input checked="" type="checkbox"/>
Dar un estimado del tiempo de implementación de cada requisito.	<input checked="" type="checkbox"/>

Tabla 5.17. Lista de chequeo para el desarrollo del documento de especificación de requisitos. Basado en [SVE01] y [SAA98].

<b>Requisitos no Funcionales</b>	
Especificar tiempos de respuesta expresados en términos de promedios, desviaciones estándar y similares.	<input checked="" type="checkbox"/>
Especificar los requisitos de carga (p.e., el sistema deberá ser capaz de gestionar 30 transacciones simultáneas).	<input checked="" type="checkbox"/>
Especificar las condiciones de tasa de transferencia (p.e., el sistema deberá procesar 30 transacciones por minuto).	<input checked="" type="checkbox"/>
Especificar las condiciones de capacidad (p.e., el sistema deberá ser capaz de almacenar 30 días de reportes de incidentes).	<input checked="" type="checkbox"/>
Especificar las condiciones de precisión y exactitud que el sistema debe cumplir y los algoritmos requeridos para implementarlas (p.e., manejo de cifras significativas, número de cifras decimales y similares).	<input checked="" type="checkbox"/>
Describir los requisitos de confiabilidad y mantenibilidad (p.e., tiempo medio entre fallos, tiempo promedio para poner a disposición, etc.).	<input checked="" type="checkbox"/>
Especificar las condiciones de seguridad que el sistema debe satisfacer (p.e., autenticación, autorización, no repudiación, etc.).	<input checked="" type="checkbox"/>
Describir el entorno hardware en el que correrá el producto.	<input checked="" type="checkbox"/>
Describir los requisitos de instalación.	<input checked="" type="checkbox"/>
Describir cualquier requerimiento relativo a documentación técnica y manuales de usuario.	<input checked="" type="checkbox"/>
<b>Interfases Internas y Externas, Incluyendo los Datos y los Controles que Fluyen a través de Ellas</b>	
Desde/a dispositivos hardware.	<input checked="" type="checkbox"/>
Desde/a dispositivos de visualización (pantallas).	<input checked="" type="checkbox"/>
Desde/a usuarios.	<input checked="" type="checkbox"/>
Desde/a otros sistemas, incluyendo sistemas legados.	<input checked="" type="checkbox"/>
Desde/a jurisdicciones externas (entidades u organizaciones externas).	<input checked="" type="checkbox"/>
Interfases entre subsistemas o componentes software.	<input checked="" type="checkbox"/>
<b>Entradas</b>	
Identificar el origen de las entradas (actores humanos u otros sistemas).	<input checked="" type="checkbox"/>
Establecer la frecuencia de llegada entre entradas.	<input checked="" type="checkbox"/>
Establecer los rangos válidos para las entradas (en el caso de datos numéricos) y las unidades de medida que definen.	<input checked="" type="checkbox"/>
Otorgar a cada entrada un número único y un identificador.	<input checked="" type="checkbox"/>

Tabla 5.17. Lista de chequeo para el desarrollo del documento de especificación de requisitos. Basado en [SVE01] y [SAA98].

<b>Salidas</b>	
Establecer las salidas en tiempo real (p.e., mensajes de alertas a un dispositivo de visualización).	<input checked="" type="checkbox"/>
Establecer las salidas que no son en tiempo real (p.e., reportes ejecutivos impresos en papel).	<input checked="" type="checkbox"/>
Identificar el destino de las salidas (dispositivos o usuarios).	<input checked="" type="checkbox"/>
Establecer los rangos válidos y las unidades de medida para las salidas.	<input checked="" type="checkbox"/>
Para cada salida definir un nombre único y un identificador.	<input checked="" type="checkbox"/>
<b>Verificación y validación de requisitos</b>	
¿Se ha designado un identificador para cada requisito para propósitos de rastreabilidad?	<input checked="" type="checkbox"/>
¿Qué justifica y motiva cada requisito? ¿Los requisitos están en concordancia con los objetivos planteados?	<input checked="" type="checkbox"/>
¿Cómo se verificará la realización de cada requisito?	<input checked="" type="checkbox"/>
Relación entre requisitos.	<input checked="" type="checkbox"/>
¿El requisito es realmente necesario? ¿Habrà una deficiencia en el producto si el requerimiento es eliminado?	<input checked="" type="checkbox"/>
¿El requisito esta reducido a su mínima expresión? ¿Puede ser dividido en otros requisitos?	<input checked="" type="checkbox"/>
Verificar que el requisito exprese realmente lo que se requiere y no que indique como debe ser alcanzado.	<input checked="" type="checkbox"/>
¿Es realmente el requisito alcanzable?	<input checked="" type="checkbox"/>
¿Se encuentra el requisito completamente especificado? ¿Requiere ampliación?	<input checked="" type="checkbox"/>
Debe verificarse que el requisito sea consistente y que no se contradiga o solape con otros requisitos. ¿Existen requisitos duplicados? ¿Se presentan especificaciones de requisitos muy parecidas o iguales?	<input checked="" type="checkbox"/>
¿Hay requisitos ambiguos?	<input checked="" type="checkbox"/>
¿Existen especificaciones de requisitos que utilicen términos vagos como "flexible" o "amigable"?	<input checked="" type="checkbox"/>
¿Es el requisito verificable por inspección, análisis, demostración o prueba? ¿Existe un criterio de verificación para cada requisito?	<input checked="" type="checkbox"/>
¿Existen criterios de priorización para los requisitos? ¿Los requisitos se encuentran priorizados?	<input checked="" type="checkbox"/>
¿Se encuentra el conjunto total de requisitos completamente especificados y no requieren de mayor profundización?	<input checked="" type="checkbox"/>

Tabla 5.17. Lista de chequeo para el desarrollo del documento de especificación de requisitos. Basado en [SVE01] y [SAA98].

### 5.3. Selección del Proveedor y Contratación

El propósito del proceso de Selección del Proveedor y Contratación es desarrollar una solicitud de propuesta que identifique y aglutine las necesidades organizacionales respecto a la adquisición del producto software, y seleccionar el proveedor que mejor pueda satisfacer los requerimientos contractuales<sup>29</sup>.

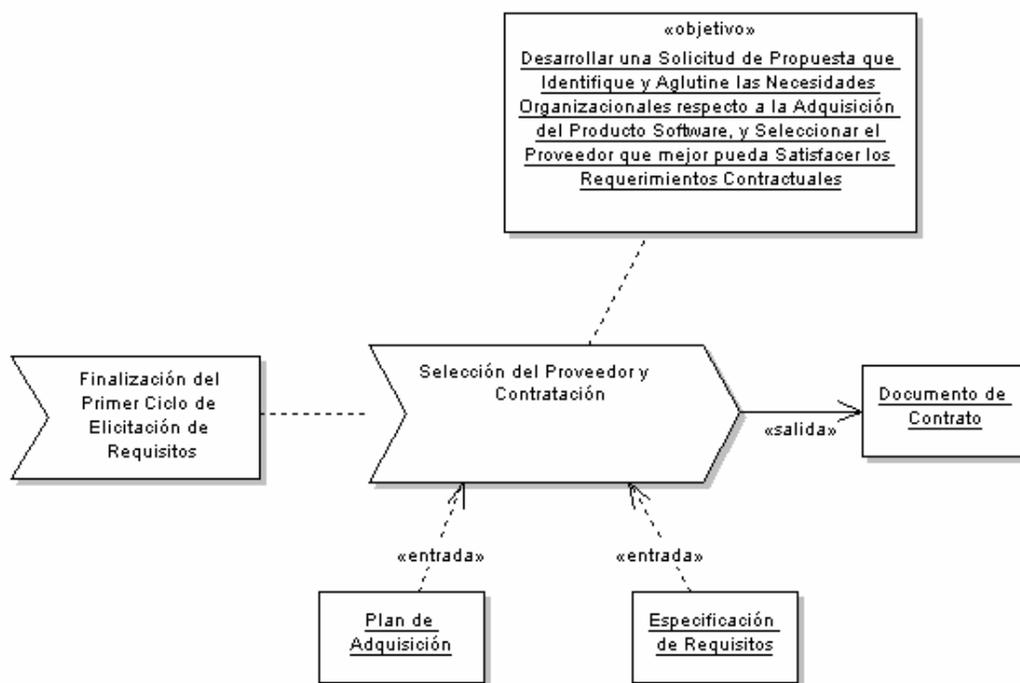


Figura 5.4. Diagrama de proceso de negocio del proceso Selección del Proveedor y Contratación.

La figura 5.4 describe los elementos esenciales que intervienen en el proceso de Selección del Proveedor y Contratación. El proceso se inicia cuando un primer conjunto de requisitos se encuentra especificado. Debe tenerse en cuenta que el proceso de desarrollo de requisitos, como cualquier otro, puede realizarse de manera iterativa e ir refinando la especificación hasta obtener una que de manera muy estable represente lo que la organización contratante

<sup>29</sup> [COF02], p. 30.

desea. Este proceso asume que una versión lo suficientemente amplia y estable de la especificación de requisitos ha sido desarrollada. De manera conjunta, este proceso se alimenta de las especificaciones descritas en el plan de adquisición (plan del proyecto) para extraer la visión global que la organización contratante tiene del proyecto, la problemática a solucionar, clarificar algunas necesidades, identificar los recursos disponibles, y establecer las limitaciones de tiempo y costos. Después de realizadas las actividades descritas en el proceso, se ha seleccionado el proveedor mejor calificado para afrontar los retos propuestos por el cliente, y se han establecido acuerdos que representan los intereses de ambos. Dichos acuerdos toman matices legales y quedan completamente descritos en el documento de contrato.

#### **5.3.1. Objetivos**

- Desarrollar un solicitud de propuesta que recoja las necesidades organizacionales
- Determinar el tipo de contrato que mejor se adecue a los intereses organizacionales
- Identificar y seleccionar el proveedor que mejor se encuentre calificado para suplir los requerimientos establecidos
- Establecer acuerdos que representen los intereses de las partes involucradas
- Desarrollar un contrato que proteja los intereses tanto de la organización contratante como del proveedor

#### **5.3.2. Actividades a Desarrollar**

- Validar las Actividades de Solicitud
- Identificar Proveedores Potenciales
- Realizar Estimaciones de Tiempo y Costos
- Desarrollar y Enviar la Solicitud de Propuesta

- Seleccionar el Proveedor
- Desarrollar y Negociar el Contrato
- Revisar de Manera Conjunta los Requerimientos y el Contrato

#### **5.3.2.1. Validar las Actividades de Solicitud**

Todas las actividades y productos de trabajo (entregables) relativos a la solicitud de propuestas deben ser revisados y validados frente a las regulaciones propias de la organización contratante y frente a las regulaciones del entorno organizacional. Respecto a las regulaciones internas de la organización contratante, se debe verificar como mínimo los siguientes aspectos:

- Políticas organizacionales respecto a la adquisición de productos y servicios.
- Normas organizacionales vigentes respecto a la adquisición de productos y servicios, y los mecanismos de contratación.
- Mecanismos de contratación permitidos.
- Montos permitidos relativos a los mecanismos de contratación: ¿hasta que monto puede ascender el contrato?
- Aspectos de confidencialidad.
- Estrategias y formas de pago a proveedores.
- Plantillas y documentos organizacionales relativos a la adquisición de productos y servicios que deben ser elaborados o tenidos en cuenta.
- Mecanismos para la proposición, evaluación y selección de proveedores.
- Estándares o regulaciones de calidad a ser tenidas en cuenta.
- Identificación del personal organizacional encargado de llevar a cabo las labores propias de la compra de productos y la contratación de servicios.

Respecto a las regulaciones relativas al entorno organizacional se debe tener en cuenta:

- Leyes nacionales relativas a la compra y subcontratación de productos y servicios.
- Entidades gubernamentales, departamentales y municipales que regulan de algún modo los procesos de adquisición y subcontratación de productos y servicios.
- Procesos y procedimientos a seguir frente a estas entidades.
- Mecanismos de revisión y validación de contratos frente a notarias y otras entidades involucradas.
- Formatos, solicitudes, plantillas y otros documentos a elaborar o a ser tenidos en cuenta frente a estas entidades.
- Normas técnicas a tener en cuenta para la elaboración de los productos de trabajo relativos al proceso de identificación, evaluación, selección y contratación de proveedores: solicitud de información a proveedores, contratos, solicitud de propuestas y similares.

Después de haber realizado una revisión de estos aspectos, debe analizarse si el proceso de identificación, evaluación, selección y contratación establecido respeta las regulaciones internas y externas a la organización. Debe tenerse en cuenta, además, que los distintos productos de trabajo generados dentro del proceso también deben satisfacer dichas regulaciones. Finalmente, es importante realizar una valoración del proceso establecido frente a los principios éticos y morales que lo deberían regir y frente a los principios organizacionales existentes en ese sentido (por ejemplo, los aspectos de confidencialidad). Cualquier situación anormal que se detecte en torno a estos principios y en torno a las regulaciones analizadas, amerita una intervención y revisión inmediata del proceso establecido.

### **5.3.2.2. Identificar Proveedores Potenciales**

Esta actividad plantea la identificación de proveedores que posiblemente pueden satisfacer las necesidades de negocio, expectativas y requerimientos que justificaron el montaje de un proyecto de adquisición. Conjuntamente, esta actividad supone un filtro a la hora de establecer proveedores potenciales ya que la búsqueda se encuentra acotada por el perfil del proveedor generado implícitamente por las necesidades y requerimientos que el proyecto de adquisición define. Dependiendo de las circunstancias particulares de cada proyecto y del nivel de formalidad definido para el mismo, se puede llevar con más disciplina esta tarea predefiniendo y documentando con más exactitud el perfil requerido para el proveedor.

A nivel metodológico, la identificación de proveedores supone el monitoreo del mercado teniendo en cuenta que la información disponible acerca de proveedores y sus capacidades continuamente esta cambiando. En este sentido, la información requerida para decidir que proveedores pueden ser más apropiados para el proyecto, continuamente se esta haciendo disponible. Adicionalmente, el monitoreo del mercado propone la pro actividad en la búsqueda de información acerca de proveedores y su incorporación y utilización en la decisión que se debe tomar en el proyecto en curso y en proyectos futuros<sup>30</sup>. A continuación, se describen brevemente algunas de las tareas que dependiendo de las características propias del proyecto de adquisición deben desarrollarse:

#### **5.3.2.2.1. Exploración del Mercado**

Se realiza una investigación de mercado para detectar proveedores y fuentes de proveedores. Bajo este contexto, el término fuentes de proveedores se utiliza para denominar las fuentes de referencia que puede ser utilizadas para detectar proveedores. Para conducir la búsqueda pueden utilizarse:

---

<sup>30</sup> [SEI02], p. 436.

- Estudios de mercado y de la industria software existentes
- Información disponible acerca de proveedores con los que ya se haya trabajado: desempeño pasado, soporte y mantenimiento posterior a la entrega del producto, viabilidad corporativa, riesgos y similares
- Rastreo en Internet
- Recurrir a asociaciones o clusters de empresas software existentes
- Indagar en otras organizaciones acerca de los proveedores que manejan
- ¡Recurrir al directorio telefónico!

#### 5.3.2.2.2. Preevaluación y Preselección de Prospectos

Teniendo como insumo la información levantada en la actividad anterior, en esta tarea se realiza una evaluación previa de los proveedores potenciales a la luz de su capacidad para satisfacer las necesidades de negocio. Esta labor puede desarrollarse tan formal o tan informalmente como se requiera. En caso que el nivel de formalidad del proyecto de adquisición sea alto y que las necesidades del mismo así lo ameriten, se debe solicitar información adicional a la previamente establecida directamente a los prospectos. Con este fin, deben desarrollarse las siguientes tareas:

- Establecer unos criterios mínimos de evaluación
- Establecer un conjunto relevante de preguntas que ayuden a filtrar la lista actual de prospectos
- Elaborar la solicitud de información (RFI, por sus siglas en inglés) con los elementos establecidos en los dos pasos anteriores
- Incluir en la solicitud de información los detalles más importantes acerca de la naturaleza del proyecto
- Preguntar explícitamente por la capacidad, disponibilidad e interés que tiene el prospecto para participar en el proceso de selección

- Enviar la solicitud de información a los prospectos y notificarlos acerca de la petición
- Recibir las respuestas formuladas por los proveedores y aplicar los criterios de filtrado previamente establecidos

Respecto a las tareas arriba descritas, es importante hacer énfasis en la capacidad, disponibilidad y, más importante aún, el interés que tenga el prospecto en participar en el proceso de selección, ya que estos factores constituyen los filtros más básicos que deben ser aplicados.

#### 5.3.2.2.3. Identificar los Riesgos Asociados a Cada Proveedor

Para todas las alternativas disponibles en cuanto a proveedores se refiere, se debe elaborar una lista de riesgos. Los riesgos representan hechos o situaciones que podrían afectar la normal ejecución del proyecto y, en este sentido, deben establecerse aquellas características del proveedor que podrían generar inconvenientes.

#### 5.3.2.3. Realizar Estimaciones de Tiempo y Costos

Básicamente, en esta tarea se trata de dar una estimación aproximada del tiempo que durará el proyecto y de los costos implicados. En la literatura pueden encontrarse diversas técnicas para la estimación de tiempos y de costos de un proyecto, algunas de ellas enfocadas al contexto de los proyectos de desarrollo de software. Dependiendo de la complejidad y de la naturaleza del proyecto, pueden utilizarse técnicas robustas como los modelos paramétricos, o técnicas más livianas como las estimaciones por analogía (p.e., se toman como referencia los tiempos y los costos de un proyecto pasado con características similares). El estudio de estas técnicas se encuentran por fuera del alcance de este proyecto, pero se pueden mencionar:

- Método descendente (“top-down, por su término en inglés): este método se encuentra basado en las características globales del proyecto. Básicamente, se trata de dividir el proyecto en las fases que empiezan al mas alto nivel para asociar y aproximar a estas los tiempos y costos implicados. Este tipo de métodos aplican a estimaciones tempranas donde solo las propiedades globales son conocidas<sup>31</sup>.
- Método ascendente (bottom-up): esta técnica involucra la identificación, el desglose y la estimación de los componentes más pequeños del proyecto, para después reunirlos todos y obtener una estimación global del mismo. Esta técnica, aunque mas acertada, requiere de un mayor esfuerzo en términos de tiempo y de recursos, y es difícil aplicarla en etapas tempranas del proyecto<sup>32</sup>.
- Estimación basada en casos de uso: esta es una técnica reciente que se basa en casos de uso para realizar las estimaciones y, por razones obvias, se encuentra muy orientada a proyectos de desarrollo de software. Básicamente, se trata de asociar una complejidad a cada caso de uso definido y de dar valor a una serie de parámetros que expresan condiciones del entorno. Para obtener información mas detallada de esta técnica puede recurrirse a [PER04].

#### **5.3.2.4. Desarrollar y Enviar la Solicitud de Propuesta**

El propósito de esta actividad es desarrollar y efectuar una solicitud formal de propuesta a los proveedores potenciales. Tan pronto la solicitud de propuesta es enviada a los proveedores y su recepción es verificada, se da por sentado que estos han sido oficialmente notificados acerca de la intención de la organización contratante de hacerlos partícipes del proceso de selección. La solicitud de propuesta (RFP, por sus siglas en inglés) expone a los proveedores interesados la naturaleza del proyecto, los requerimientos del mismo, la forma en que se deben desarrollar las propuestas, como serán estas evaluadas y la

---

<sup>31</sup> [GIC95], p. 119.

<sup>32</sup> [GIC95], p. 119.

fecha límite de entrega. La tabla 5.18 describe los principales elementos que debe contener una solicitud de propuesta.

<b>Descripción General de la Organización Contratante</b>	
La organización contratante se identifica o da a conocer ante los proveedores a través de elementos tales como la misión, visión, razón del negocio y objetivos corporativos. Es importante, además, proporcionar información relativa al tamaño de la organización, número de empleados, número de sedes, ubicación y distribución geográfica, teléfonos y contactos.	<input checked="" type="checkbox"/>
<b>Visión y Objetivos del Proyecto</b>	
Los objetivos del proyecto deben ser descritos de forma clara y precisa. Cada objetivo debe estar asociado a una o más necesidades de negocio y no ha aspectos específicos, atributos o características de la solución software requerida. Adicionalmente, se debe describir de forma breve la visión de clientes y usuarios acerca de la solución ambicionada.	<input checked="" type="checkbox"/>
<b>Dominio del Problema y Entorno del Producto</b>	
La situación por la cual se genera la necesidad organizacional de desarrollar un producto software debe ser descrita en términos de necesidades de negocio. Es importante también describir los aspectos relevantes al entorno donde se espera que el producto interactúe. Estos aspectos pueden referirse a las dependencias organizacionales involucradas en el problema, procesos de negocio, procesos de soporte, cargos y funciones, afectados y similares.	<input checked="" type="checkbox"/>
<b>Entregables del Proyecto</b>	
Se debe desarrollar de la forma más precisa posible una lista de los entregables que se espera que el proyecto arroje. Los entregables pueden referirse planes del proyecto, código fuente, actas, plantillas, bases de datos y, en general, cualquier tipo de documentación que se requiera.	<input checked="" type="checkbox"/>
<b>Requerimientos del Proyecto</b>	
El conjunto total de requerimientos establecidos para el proyecto de adquisición deben ser expuestos a los proveedores potenciales ya sea como anexos o como parte del contenido de la solicitud de propuesta. Los requerimientos pueden hacer referencia a los requisitos cliente, requerimientos de soporte y mantenimiento, requerimientos de documentación, requerimientos de calidad, requerimientos de capacitación o similares.	<input checked="" type="checkbox"/>
<b>Especificaciones Contractuales</b>	
En este apartado se deben identificar y documentar las restricciones contractuales aplicables al proveedor que sean más relevantes para la organización contratante. Las restricciones relativas a la reserva, confidencialidad y seguridad de la información se consideran críticas y, por tanto, siempre deben ser exhibidas en las solicitudes de propuesta. Adicionalmente, el tipo de contrato que se pretende elaborar con el proveedor también debe ser especificado (ver tabla 5.23).	<input checked="" type="checkbox"/>

Tabla 5.18. Lista de chequeo para la solicitud de propuesta (RFP, por sus siglas en inglés). Basado en [SVE01].

<b>Estimados de Tiempo y Costos</b>	
Con el fin que el proveedor se haga una idea del presupuesto disponible para el proyecto y de las limitaciones de tiempo existentes, es importante incluir en la solicitud de propuesta los estimados de costos y plazos realizados.	<input checked="" type="checkbox"/>
<b>Disposición de Recursos</b>	
Los recursos que la organización contratante pone a disposición del proyecto como equipos, hardware, software, oficinas y similares, deben ser especificados. De manera similar, los recursos que se espera que el proveedor ponga a disposición del proyecto también deben ser descritos. Lo anterior se aplica igualmente a los recursos humanos involucrados.	<input checked="" type="checkbox"/>
<b>Instrucciones para la Elaboración de la Contrapropuesta</b>	
Es importante que en la solicitud de propuesta se plasmen las consideraciones que los proveedores prospecto deben guardar en la elaboración de la contrapropuesta. La contrapropuesta, también denominada respuesta a la propuesta, debe contener como mínimo los siguientes aspectos:	
· Descripción de la compañía, recorrido y experiencia.	<input checked="" type="checkbox"/>
· Clientes actuales y anteriores. Resultados obtenidos en otros proyectos.	<input checked="" type="checkbox"/>
· Posición financiera.	<input checked="" type="checkbox"/>
· Entendimiento del problema, solución sugerida, enfoque técnico, descripción de cómo los distintos requerimientos serán atacados.	<input checked="" type="checkbox"/>
· Habilidades técnicas y de gestión.	<input checked="" type="checkbox"/>
· Proceso de desarrollo de software y métodos de trabajo.	<input checked="" type="checkbox"/>
· Prácticas de calidad.	<input checked="" type="checkbox"/>
· Recursos de los que dispone: equipos y herramientas.	<input checked="" type="checkbox"/>
· Cumplimiento de estándares (p.e., CMMI).	<input checked="" type="checkbox"/>
· Estimados de tiempo y costos.	<input checked="" type="checkbox"/>
· Gestión de riesgos y estrategias de mitigación.	<input checked="" type="checkbox"/>
· Personal con el que cuenta; organización y constitución del personal; experiencia del personal.	<input checked="" type="checkbox"/>
· Contactos, teléfonos, direcciones y correos.	<input checked="" type="checkbox"/>
· Precios, honorarios, cargos y asuntos relativos al pago.	<input checked="" type="checkbox"/>
· Uso de subcontratistas, ¿quiénes son?, ¿para qué los utiliza?	<input checked="" type="checkbox"/>
· Inversiones requeridas en equipos, herramientas y personal.	<input checked="" type="checkbox"/>
· Aspectos legales relativos a garantías, licencias, derechos de autor, propiedad intelectual y similares. En el proceso de Transición a Operaciones y Soporte (tabla 5.21) se ofrece una lista de chequeo que ayuda a clarificar estos temas.	<input checked="" type="checkbox"/>
<b>Criterios y Mecanismos de Evaluación</b>	
Es importante que los proveedores potenciales tengan conocimiento acerca de los criterios y mecanismos de evaluación establecidos por la organización contratante. Lo anterior, sirve para minimizar el riesgo de malos entendidos en el proceso de selección y ayuda a que los proveedores se preparen mejor para el mismo.	<input checked="" type="checkbox"/>

Tabla 5.18. Lista de chequeo para la solicitud de propuesta (RFP, por sus siglas en inglés). Basado en [SVE01].

<b>Derechos de Inspección de la Organización Contratante</b>	
En la solicitud de propuesta deben incluirse los derechos que la organización contratante pretende obtener para inspeccionar características del proveedor tales como la posición financiera, las capacidades técnicas, la experiencia, las prácticas de calidad, las certificaciones obtenidas y similares.	<input checked="" type="checkbox"/>
<b>Descripción de Aspectos Legales</b>	
En este apartado deben identificarse todas las consideraciones legales que incumben tanto a la organización contratante como a los proveedores prospecto. Dichas consideraciones pueden referirse a cláusulas de garantías, licencias, derechos de propiedad, derechos de autor, patentes, derechos de uso y otros temas relacionados con la propiedad intelectual.	<input checked="" type="checkbox"/>
<b>Utilización de Terceros</b>	
La solicitud de propuesta debe especificar las restricciones relacionadas a la utilización de terceros y de subcontratistas. Debe especificarse, además, los mecanismos de control a aplicar para adelantar las labores de gestión de otros proveedores.	<input checked="" type="checkbox"/>
<b>Descripción de Riesgos</b>	
Con el fin que los proveedores prospecto se enteren de los temas con los cuales posiblemente deban tratar, es importante que en la solicitud de propuesta se incluyan los riesgos que, hasta esta etapa del proyecto, se han identificado. De igual modo, deben incluirse las estrategias de mitigación y de contingencia que hasta el momento se hayan desarrollado.	<input checked="" type="checkbox"/>

Tabla 5.18. Lista de chequeo para la solicitud de propuesta (RFP, por sus siglas en inglés). Basado en [SVE01].

### 5.3.2.5. Seleccionar el Proveedor

Como se mencionó anteriormente, la escogencia del proveedor o de los proveedores que adelantarán las labores de desarrollo es uno de los aspectos más críticos y determinantes para el éxito del proyecto. La selección de un proveedor debe estar debidamente fundamentada en criterios de evaluación previamente establecidos. Los criterios de evaluación son pautas que ayudan a la organización contratante a valorar las solicitudes de propuesta diligenciadas por los proveedores potenciales y, básicamente, sirven a dos propósitos:

- Habilitar a los participantes del proyecto para la estandarización de los criterios a ser considerados en la evaluación de propuestas.

- Facilitar el entendimiento de los proveedores potenciales de la forma en que las propuestas serán revisadas, tanto individualmente como en comparación con otras propuestas.

#### 5.3.2.5.1. Características de los Criterios de Evaluación

La tabla 5.19 describe algunas de las características que los criterios de evaluación deben de satisfacer con el fin que sean realmente efectivos:

<b>Característica / Objetivo</b>	<b>Propósito</b>
Objetivos	No susceptibles a diversas interpretaciones.
Relativos a los requerimientos definidos	Los requerimientos clave del proyecto deben ser cubiertos por los criterios de evaluación.
Discriminantes	Separan las mejores propuestas, las propuestas promedio y las propuestas más débiles.
No discriminantes	Imparciales y razonables.
Realistas	Se adecuan a la naturaleza del contrato.
Medibles	Utilizan estándares medibles y subcriterios si estos sirven para simplificar la evaluación.
Económicos	No consumen una cantidad desproporcionada de tiempo y recursos.
Justificables	Pueden ser justificados por simple sentido común o por argumentos técnicos o legales.

Tabla 5.19. Características de los criterios de evaluación. Basado en [ASN97].

Los criterios de evaluación deben ser claros y tener un propósito para que tanto el equipo del proyecto como los proveedores potenciales entiendan la forma en que será valorada la propuesta. Conjuntamente, estos deben estar en concordancia con los requerimientos que el proyecto define, con la naturaleza del contrato y con los recursos disponibles. Por último, es importante que los criterios de evaluación que se definan ayuden a identificar las mejores propuestas, para lo cual estos deben definir algún mecanismo de medición. Los criterios de evaluación que no sean medibles ni cuantitativa ni cualitativamente, no deben ser tenidos en cuenta.

#### 5.3.2.5.2. Tipología de los Criterios de Evaluación

Los criterios de evaluación para un proyecto de adquisición pueden organizarse de acuerdo a una infinidad de aseveraciones, pero en general estos caben en alguna de las siguientes categorías:

##### 5.3.2.5.2.1. Criterios Técnicos

Los criterios técnicos son utilizados para evaluar el mérito o la excelencia del enfoque técnico o la metodología propuesta por el proveedor potencial. De igual modo, estos criterios también sirven para valorar las tareas a ser desarrolladas o el producto software a ser proveído. Algunos de los criterios técnicos que pueden considerados son:

- Entendimiento del problema
- Entendimiento de los requisitos cliente y de producto
- Habilidad para satisfacer los requerimientos establecidos
- Enfoque metodológico y de la solución
- Identificación de áreas de problemas críticas (riesgos)
- Análisis de riesgos
- Metodología de pruebas
- Personal técnico
- Capacidad de servicio y de soporte
- Organización de las actividades del proyecto o de la forma de trabajo

##### 5.3.2.5.2.2. Criterios Referentes a los Costos

Los criterios de costos son utilizados para evaluar el costo asociado a las propuestas realizadas por los proveedores potenciales. Los costos generalmente son evaluados en términos de los siguientes criterios:

- Costo estimado del proyecto y de los honorarios
- Costo total del ciclo de vida del proyecto
- Variabilidad de los costos
- Mecanismos para el control de costos
- Consistencia de los costos con los distintos planes del proyecto
- Realismo de los costos propuestos
- Sensatez

#### 5.3.2.5.2.3. Criterios Referentes a la Gestión de Proyectos

Los criterios referentes a la gestión de proyectos o simplemente criterios de gestión, son empleados para valorar la habilidad del proveedor para desempeñarse correctamente a través del ciclo de vida del proyecto. Dicha habilidad puede ser valorada en términos de los siguientes criterios:

- Experiencia
- Desempeño pasado
- Planes propuestos para manejar el proyecto
- Recursos (tanto humanos como materiales)
- Actitud y respuesta del proveedor
- Enfoque de gestión del proyecto
- Enfoque de gestión de las relaciones
- Utilización de otros proveedores
- Planificación del proyecto

A continuación se describen algunas de las actividades que deben desarrollarse a la hora de evaluar y seleccionar proveedores.

#### 5.3.2.5.3. Determinar los Criterios de Evaluación

La escogencia de criterios para la evaluación de proveedores potenciales depende netamente de las características propias de cada proyecto. La naturaleza de la adquisición, el grado de formalidad del proyecto, el tamaño y tipo del producto software ambicionado por clientes y usuarios, los recursos disponibles, el nivel de complejidad estimado del sistema y similares, son factores que influyen en la definición de los criterios de evaluación. Es importante tener en cuenta que los criterios de evaluación que se escojan deben reflejar con precisión los requerimientos contractuales. Debe tenerse en cuenta, además, que el establecimiento de los criterios de evaluación es una labor crítica y, como se dijo anteriormente, es determinante en la escogencia del proveedor que mejor satisface una necesidad en particular.

#### 5.3.2.5.4. Establecer la Relevancia de los Criterios de Evaluación

Una vez definidos los criterios de evaluación, se debe establecer la importancia relativa a cada uno de ellos. Cuando se habla de importancia, se esta hablando de la criticidad o del impacto que el criterio de evaluación define para el éxito en la escogencia del proveedor que mejor podría desempeñarse durante todas las fases del ciclo de vida del proyecto. Al igual que con la escogencia de los criterios de evaluación, determinar la relevancia de estos depende de las circunstancias y necesidades de la organización contratante. En este sentido, es la organización contratante la responsable de establecer la escala con la cual se juzgarán cada una de los criterios de evaluación. Además, debe considerarse que la importancia establecida para cada criterio de evaluación y la forma como esta se mida, tiene impacto directo tanto en el mecanismo de evaluación utilizado como en los resultados del mismo, es decir, entre más importancia se le otorgue a cierto factor más se favorecerá a los proveedores potenciales que lo satisfagan.

#### 5.3.2.5.5. Definir las Técnicas de Valoración de Criterios

Como se dijo anteriormente, cada criterio de evaluación debe tener un valor asignado que indique la importancia que para la organización contratante tiene ese criterio y que le permita realizar mediciones sobre las propuestas elaboradas por los proveedores potenciales. Por lo general, los valores asignados pueden ser tanto cuantitativos como cualitativos, sin embargo, no existen técnicas prescriptivas que indiquen como deben ser calificados los criterios de evaluación. A continuación, se listan algunas de las técnicas de calificación de criterios que podrían ser utilizadas:

- Códigos o convenciones de colores
- Asignación de caracteres como el “+” a los criterios mejor satisfechos, “x” a los regularmente satisfechos y “-” a los pobremente satisfechos.
- Listas de chequeo de criterios
- Alguna escala numérica de calificación de criterios
- Alguna escala cualitativa de calificación de criterios (se utilizan adjetivos)

Sin importar que técnica se utilice, es importante que la calificación sea acompañada de una descripción textual que indique las fortalezas, debilidades, mérito y riesgos asociados a cada uno de los factores evaluados. Si la técnica de calificación empleada es numérica, el equipo del proyecto debe realizar un referencia cruzada entre la escala numérica y la escala cualitativa (representada por adjetivos); lo anterior, con el propósito de asegurar la uniformidad en las calificaciones otorgadas y de proveer un balance entre estas y las descripciones textuales realizadas.

Respecto a la clasificación de los criterios de evaluación, es recomendable que los criterios referentes a la gestión de proyectos sean calificados con técnicas diferentes a las numéricas. Lo anterior, es debido a la naturaleza adjetiva de este tipo de criterios y al mayor grado de objetividad y de claridad que se puede

alcanzar utilizando una escala cualitativa. Adicionalmente, es fundamental que la valoración de los criterios referentes a costos se realice al final y no al principio del proceso de evaluación. Esto es muy importante, ya que para tener una visión objetiva de los costos asociados a una propuesta primero se debe entender lo que el proveedor potencial realmente está ofreciendo, es decir, se debe haber valorado previamente las capacidades del mismo. Finalmente, a pesar que los criterios referentes a costos y a la gestión de proyectos usualmente no son valorados a través de técnicas numéricas, es importante entender que estos hacen parte integral y determinante del proceso de evaluación de proveedores y, en este sentido, deben ser juiciosamente contemplados.

#### 5.3.2.5.6. Efectuar la Evaluación

Teniendo en cuenta la complejidad del proyecto de adquisición y la cantidad de proveedores potenciales a ser evaluados, es importante que la técnica de evaluación que se emplee sea la más simple posible. Como se mencionó anteriormente, los criterios referentes a costos y a la gestión de proyectos usualmente son valorados de forma cualitativa. Lamentablemente, este tipo de valoración es susceptible a prejuicios personales y puede dar cabida a subjetividades indeseables en la evaluación de proveedores. Adicionalmente, elaborar una técnica de evaluación aparte para los elementos cualitativos puede agregar mucha complejidad al proceso de evaluación. Dado que los criterios de costos y de gestión no deben ser desconocidos y con el fin de minimizar el impacto de los prejuicios personales y de simplificar el proceso de selección, se recomienda utilizar algún sistema de ponderación numérica para dichos criterios [HUW06].

A continuación, se describe una técnica de evaluación de proveedores que va en sintonía con las recomendaciones arriba descritas. Esta técnica debe desarrollar las siguientes tareas:

1. Asignar un peso numérico a cada criterio de evaluación.
2. Definir fórmulas o mecanismos numéricos para calcular la calificación de cada criterio de evaluación.
3. Calificar a los proveedores potenciales en cada criterio.
4. Totalizar las calificaciones obtenidas por cada proveedor.

La tabla 5.20 esquematiza la propuesta para la evaluación de proveedores y agrupa el resultado de las tareas arriba mencionadas.

Tipo Criterio	Criterio de Evaluación	Fórmula / Método de Cálculo	Peso	Proveedor 1	...	Proveedor n...
<i>Técnicos</i>	Criterio 1.1					
	⋮					
	Criterio 1.n					
<i>Referentes a la Gestión de Proyectos</i>	Criterio 2.1					
	⋮					
	Criterio 2.n					
<i>Referentes a Costos</i>	Criterio 3.1					
	⋮					
	Criterio 3.n					
<b>Puntaje Total:</b>			$\Sigma$ Pesos	$\Sigma$ Puntajes		$\Sigma$ Puntajes

Tabla 5.20. Esquema para la evaluación de proveedores.

#### 5.3.2.5.6.1. Tipo de Criterio

Para efectos de una mejor organización, este atributo clasifica o agrupa los criterios de evaluación de acuerdo a la tipología de los mismos: criterios técnicos, criterios de gestión o criterios referentes a costos. Si el equipo del proyecto identifica y requiere alguna otra clasificación, esta puede ser adicionada.

#### 5.3.2.5.6.2. Criterio de Evaluación

Como se mencionó anteriormente, los criterios de evaluación son factores que ayudan a la organización contratante a valorar las solicitudes de propuesta diligenciadas por los proveedores potenciales. Dichos criterios deben ser plasmados en esta columna de acuerdo a su clasificación

#### 5.3.2.5.6.3. Fórmula o Método de Cálculo

Cada criterio de evaluación debe definir una fórmula o algún mecanismo que permita calificar el grado de satisfacción de dicho criterio por un proveedor en particular. El mecanismo empleado por los criterios cuantitativos es totalmente dependiente de la naturaleza propia de cada criterio, es decir, cada criterio define sus propias variables y la función matemática que lo calcula. Para los criterios cualitativos es posible emplear alguna de las siguientes técnicas:

##### 5.3.2.5.6.3.1. Puntajes Indexados

Se elabora una tabla de dos columnas para el criterio cualitativo. En la segunda columna se define la escala de puntuación que se utilizará para calificar el criterio. Se debe tener en cuenta que la puntuación máxima que se asigne a esta escala esta supeditada al peso que se le haya asignado al criterio. En la primera columna se definen los factores, fórmulas o asecciones que hacen que el criterio de evaluación obtenga la calificación especificada en la segunda columna. La tabla 5.21 muestra un ejemplo de esta técnica:

<b>Criterio: Fácil integración con los sistemas existentes.</b>	
<b>Asecciones</b>	<b>Puntuación</b>
Fácil Integración	5
Se requiere poco trabajo para realizar la integración (<= 1 semana)	4

Tabla 5.21. Ejemplo de la técnica de Puntajes Indexados. Basado en [HUW06].

Se requieren entre 1 y 3 semanas para realizar la integración.	3
Se requiere mucho trabajo para realizar la integración (> 3 semanas).	2
Los sistemas existentes deberán ser reemplazados para acomodar el nuevo sistema.	1
Difícilmente la integración pueda ser realizada.	0

Tabla 5.21. Ejemplo de la técnica de Puntajes Indexados. Basado en [HUW06].

### 5.3.2.5.6.3.2. Ponderación de los Criterios de Evaluación

Se elabora una tabla de cuatro columnas para el criterio de evaluación que se tenga en cuestión. En la primera columna se especifican los subcriterios o subcaracterísticas aplicables al criterio cualitativo. En la segunda columna se especifica el peso asociado a cada subcaracterística. Se debe tener en cuenta que la sumatoria de los pesos asignados a las subcaracterísticas debe ser igual al peso asignado al criterio de evaluación. En la tercera columna se asigna una calificación a la subcaracterística relacionada: uno si se satisface, o cero en caso contrario. En la columna de puntaje se plasma el resultado de la multiplicación entre el peso y la calificación obtenida por la subcaracterística. El puntaje total asociado al criterio se obtiene sumando los valores contenidos en la columna de calificación. La tabla 5.22 muestra un ejemplo de esta técnica:

Capacidad del Proceso	Peso	Calificación	Puntaje
Calidad del Software	1	1	1
Gestión del Proyecto	2	1	2
Gestión de la Configuración	1	1	1
Administración de Requisitos	1	0	0
Ingeniería de Sistemas e Ingeniería de Software	2	1	2
Verificación y Validación	2	1	2
Gestión de Riesgos	1	0	0
<b>Puntaje Total:</b>			<b>8</b>

Tabla 5.22. Ejemplo de la técnica de ponderación de los criterios de evaluación. Basado en [HUW06].

#### 5.3.2.5.6.4. Peso

El peso es un valor numérico que se le asigna a cada criterio y representa la importancia que este tiene para la organización contratante en el proceso de selección. El peso también puede ser visto como el máximo puntaje que un proveedor puede obtener frente a cierto criterio, o como una ponderación del mismo. La escogencia de una escala para definir los pesos se realiza a criterio de la organización contratante.

#### 5.3.2.5.6.5. Proveedor

En la columna proveedor se plasma el resultado que el proveedor obtiene frente al criterio de evaluación, es decir, su calificación. Dicha calificación es el resultado de la fórmula o mecanismo de cálculo definido para el criterio de evaluación. Debe tenerse en cuenta que los criterios referentes a costos deben ser calculados de último ya que su correcta interpretación depende de la propuesta técnica y de gestión realizada por el proveedor.

#### 5.3.2.5.6.6. Puntaje Total

Esta fila representa el puntaje total obtenido por el proveedor en la evaluación de los criterios definidos. Para calcular el puntaje total de un proveedor deben sumarse las calificaciones obtenidas por este en cada uno de los criterios de evaluación. El anterior procedimiento debe realizarse para cada proveedor que se encuentre en la tabla y también para la columna de pesos.

#### 5.3.2.5.7. Definición del Proveedor y Tareas Adicionales

Esta es la parte del proceso de selección de proveedores en donde se realiza una recopilación de los resultados obtenidos en la evaluación, se determina el

mayor puntaje y se escoge al proveedor. El proveedor seleccionado debe ser avisado oportunamente acerca de la elección para darle el tiempo requerido para que se prepare para las actividades de contratación. De manera similar, y por cuestiones de cortesía, los demás proveedores que participaron en el proceso de selección también deben ser notificados acerca de la elección realizada, con el fin que puedan liberar tranquilamente los recursos dispuestos al mencionado proceso.

#### **5.3.2.6. Desarrollar y Negociar el Contrato**

Después que el proveedor ha sido seleccionado un contrato de negocios debe ser establecido entre las partes. Dada la inherente importancia de esta actividad y el grado de especialización requerido para desarrollarla, es imprescindible contar con una buena asistencia legal. En este sentido, debe recurrirse al departamento jurídico de la organización contratante para que los contratos que se elaboren sean minuciosamente revisados por abogados. Las tareas más importantes que debe tenerse en cuenta a la hora de desarrollar un contrato son:

- Establecer el tipo de contrato a ser aplicado y desarrollar un contrato apropiado
- Determinar la forma en que los pagos serán realizados
- Establecer las obligaciones contractuales de las partes involucradas
- Incorporar los requerimientos del proyecto y los criterios de evaluación en el contrato
- Incorporar los procedimientos requeridos para desarrollar las pruebas de aceptación

A continuación se describen los aspectos más importantes de cada una de estas tareas:

#### 5.3.2.6.1. Establecimiento del Tipo de Contrato

Seleccionar el tipo de contrato que mejor represente los requerimientos del proyecto es una de las decisiones más importantes en el proceso de adquisición. Básicamente, existen dos tipos de contratos a ser tenidos en cuenta en un proyecto de desarrollo de software: contrato de costos unitarios fijos y contrato de costos reembolsables. En el contrato de costos unitarios fijos el proveedor asume la mayor parte de los riesgos asociados a los costos del proyecto. Debido a esto, los requerimientos del proyecto y su alcance necesitan ser estrictamente definidos y los cambios que se puedan presentar deben ser manejados con rigurosidad. Es imprescindible, además, que se evalúe concienzudamente cuán alcanzables son los objetivos que el proyecto define en aras de minimizar el riesgo que se presenten conflictos entre las partes y de incurrir en sobrecostos. En contraposición, en los contratos de costos reembolsables la organización contratante asume la mayor parte de los riesgos asociados a los costos del proyecto. Con el fin de mitigar dichos riesgos, la organización contratante debe ser muy disciplinada en el manejo de los costos generados por el proyecto y debe tener estricto control del progreso del mismo. Lo anterior genera una carga administrativa alta para la organización pero se previenen conflictos y sobrecostos. Adicionalmente, con los contratos de costos reembolsables la organización contratante adquiere una mayor flexibilidad para manejar las actividades del proyecto y para introducir cambios en el alcance, lo cual es apropiado para proyectos novedosos o para proyectos donde el alcance es volátil.

Existen ligeras variaciones sobre los tipos de contratos arriba descritos. La tabla 5.23 resume las características más importantes de los contratos aplicables a proyectos de desarrollo de software y asocia el nivel de riesgo relativo a costos en los que pueden incurrir las partes involucradas:

<b>Tipo</b>	<b>Descripción</b>	<b>Aplicabilidad</b>	<b>Administración</b>	<b>Riesgo</b>
<u>CFU - Costos Fijos Unitarios</u>	El cliente paga un precio fijo por el trabajo del proveedor.	Proyectos donde: · El costo estimado es confiable · El alcance esta bien definido y es estable · Los objetivos son alcanzables	· Poca: si se paga todo al final · Moderada: si se realizan pagos periódicos · Alta: si se paga por cumplimiento de los hitos del proyecto o por los entregables generados	 Cliente      Proveedor
<u>CFAE - Costos Fijos con Ajustes Económicos</u>	Un contrato de costos fijos con ajustes de precios por conceptos de viáticos, materiales y similares.	Proyectos donde: · El alcance esta bien definido y es estable · El contrato es extenso en tiempo y las partes necesitan protegerse de las fluctuaciones del mercado.	· Moderada: si se paga todo al final · Alta: si se realizan pagos periódicos	 Cliente      Proveedor
<u>CFUR - Costos Fijos Unitarios Redeterminables</u>	Un costo fijo inicial es negociado. Se realiza un acuerdo previo entre las partes acerca de las modificaciones que causarán aumentos en los precios: aumento de los requerimientos pactados, adición de entregables, etc.	Proyectos donde: · El alcance esta parcialmente bien definido · El alcance tiene algún grado de volatilidad	· Moderada	 Cliente      Proveedor

Tabla 5.23 Tipología de los contratos aplicables a la delegación de proyectos de desarrollo de software. Basado en [SVE01].

Tipo	Descripción	Aplicabilidad	Administración	Riesgo
<p><u>CFCV - Costos Fijos más Costos Variables</u></p>	<p>Adicionalmente al costo fijo, se determinan pagos variables negociando los siguientes parámetros:</p> <ul style="list-style-type: none"> <li>· Costo objetivo y costo límite</li> <li>· Costo objetivo mínimo y máximo</li> <li>· Fórmula de ajuste</li> </ul> <p>La fórmula de ajuste específica como el costo objetivo será compartido entre el cliente y el proveedor.</p>	<p>Proyectos donde:</p> <ul style="list-style-type: none"> <li>· El costo estimado es confiable</li> <li>· El alcance esta bien definido y es estable</li> <li>· Los objetivos son alcanzables</li> <li>· Se requiere motivar al proveedor para que incremente su eficiencia, planifique a corto plazo y reduzca costos</li> </ul>	<ul style="list-style-type: none"> <li>· Moderada: si se paga todo al final</li> <li>· Alta: si se realizan pagos periódicos</li> </ul>	 <p>Cliente      Proveedor</p>
<p><u>CFI - Costos Fijos más Incentivos</u></p>	<p>Se pacta un costo fijo con el proveedor más una utilidad básica. Adicionalmente, el proveedor recibe incentivos económicos cuando ciertos criterios son alcanzados. Los incentivos iniciales deben ser bajos (&lt; 3% del costo objetivo) e ir incrementándose gradualmente (sin sobrepasar el 20% del costo objetivo).</p>	<p>Proyectos donde:</p> <ul style="list-style-type: none"> <li>· El costo estimado es confiable</li> <li>· El alcance esta bien definido y es estable</li> <li>· Los objetivos son alcanzables</li> <li>· Se requiere estimular al proveedor para que mejore su desempeño en áreas que son difíciles de medir objetivamente: satisfacción del usuario, calidad del software, uso de mejores prácticas, etc.</li> </ul>	<ul style="list-style-type: none"> <li>· Alta</li> </ul>	 <p>Cliente      Proveedor</p>

Tabla 5.23 Tipología de los contratos aplicables a la delegación de proyectos de desarrollo de software. Basado en [SVE01].

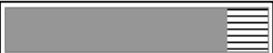
<b>Tipo</b>	<b>Descripción</b>	<b>Aplicabilidad</b>	<b>Administración</b>	<b>Riesgo</b>
<u>CC - Costos Compartidos</u>	En un contrato de costos compartidos el cliente y el proveedor acuerdan la proporción en la cual los costos serán compartidos. Usualmente, el cliente paga todos los costos permisibles. El proveedor no recibe pagos en ninguno de los otros tipos de contrato.	Proyectos donde: <ul style="list-style-type: none"> <li>· Tanto el cliente como el proveedor comparten los esfuerzos de desarrollo</li> <li>· Ambas partes comparten el riesgo</li> <li>· Ambas partes se verán beneficiadas con el producto</li> <li>· alguna de las partes es una organización sin ánimo de lucro</li> </ul>	Alta	<ul style="list-style-type: none"> <li>· 100% del lado del cliente por los costos del contrato</li> <li>· Compartido, de acuerdo con la proporción pactada para la compartición de los costos del contrato.</li> </ul>
<u>CRCF - Costos Reembolsables más Costo Fijo</u>	Los costos permitidos al proveedor son reembolsados y el precio fijo pactado es liberado después de la entrega del producto (software, documentación técnica y documentación de usuario).	Proyectos donde: <ul style="list-style-type: none"> <li>· El costo no puede ser razonablemente estimado.</li> <li>· Existen muchos elementos muy innovadores (poca experiencia)</li> <li>· El alcance es volátil o difícil de definir</li> </ul>	· Alta	 Cliente      Proveedor
<u>CRPP - Costos Reembolsables más Pagos Porcentuales</u>	Los costos permitidos al proveedor son reembolsados y un porcentaje de los mismos es recibido como pago.	Proyectos donde: <ul style="list-style-type: none"> <li>· El costo no puede ser razonablemente estimado.</li> <li>· Existen muchos elementos muy innovadores (poca experiencia)</li> <li>· El alcance es volátil o difícil de definir</li> </ul>	· Alta	 Cliente      Proveedor

Tabla 5.23 Tipología de los contratos aplicables a la delegación de proyectos de desarrollo de software. Basado en [SVE01].

Tipo	Descripción	Aplicabilidad	Administración	Riesgo
<p><u>CRCV - Costos Reembolsables más Costos Variables</u></p>	<p>Adicional al reembolso de los costos pactados con el proveedor, se determinan pagos variables negociando los siguientes parámetros:</p> <ul style="list-style-type: none"> <li>· Costo objetivo y costo límite</li> <li>· Costo objetivo mínimo y máximo</li> <li>· Fórmula de ajuste</li> </ul> <p>La fórmula de ajuste específica como el costo objetivo será compartido entre el cliente y el proveedor.</p>	<p>Proyectos donde:</p> <ul style="list-style-type: none"> <li>· El costo no puede ser razonablemente estimado.</li> <li>· Existen muchos elementos muy innovadores (poca experiencia)</li> <li>· El alcance es difícil de definir</li> <li>· Se requiere motivar al proveedor para que incremente su eficiencia, planifique a corto plazo y reduzca costos</li> </ul>	<p>· Muy alta</p>	 <p>Cliente      Proveedor</p>
<p><u>CRI - Costos Reembolsables más Incentivos</u></p>	<p>Se reembolsa al proveedor los costos pactados más una utilidad básica. Adicionalmente, el proveedor recibe incentivos económicos cuando ciertos criterios son alcanzados. Los incentivos iniciales deben ser bajos (&lt; 3% del costo objetivo) e ir incrementándose gradualmente (sin sobrepasar el 20% del costo objetivo).</p>	<p>Proyectos donde:</p> <ul style="list-style-type: none"> <li>· El costo no puede ser razonablemente estimado.</li> <li>· Existen muchos elementos innovadores (poca experiencia)</li> <li>· El alcance es difícil de definir</li> <li>· Se requiere estimular al proveedor para que mejore su desempeño en áreas que son difíciles de medir objetivamente: satisfacción del usuario, calidad del software, uso de mejores prácticas, etc.</li> </ul>	<p>· Muy alta</p>	 <p>Cliente      Proveedor</p>

Tabla 5.23 Tipología de los contratos aplicables a la delegación de proyectos de desarrollo de software. Basado en [SVE01].

Tipo	Descripción	Aplicabilidad	Administración	Riesgo
<u>CMHL - Costos por Materiales y Horas Laboradas</u>	Se utiliza una escala horaria para calcular el tiempo que el proveedor utiliza para llevar a cabo las tareas pactadas. Los costos generados por este tiempo y por los recursos materiales que el proveedor requiera para desempeñar su trabajo, son reembolsados a este.	Proyectos donde: <ul style="list-style-type: none"> <li>· La extensión y duración del trabajo no puede ser razonablemente estimada</li> <li>· Se requiere cambiar rápidamente la dirección del trabajo y llevar un monitoreo cercano</li> </ul> <i>Nota.</i> Un precio límite debe ser establecido en este tipo de contratos.	<ul style="list-style-type: none"> <li>· Poca</li> </ul>	 Cliente      Proveedor

Tabla 5.23 Tipología de los contratos aplicables a la delegación de proyectos de desarrollo de software. Basado en [SVE01].

#### 5.3.2.6.2. Determinar la Forma en que los Pagos serán Realizados

En esta tarea se establecen las obligaciones contractuales de las partes respecto a la forma en que los pagos serán liberados. En este sentido, elementos tales como las condiciones de pago, montos, costos y gastos permitidos, cargos y similares deben ser estrictamente definidos. Es importante, además, establecer como y cuando los pagos estarán atados a la liberación de entregables o al cumplimiento de los hitos definidos en el proyecto. De igual modo, variables del entorno relativas a impuestos, tasas de interés, trámites legales, tasas de cambio y afines deben ser consideradas.

Como se describe en la tabla 5.23, dependiendo del tipo de contrato a ser aplicado la forma en que se realizan los pagos puede variar. Para contratos donde se recurra al uso de incentivos (CFI y CRI), se deben especificar rigurosamente los criterios bajo los cuales los incentivos serán entregados al proveedor. Algunos de estos criterios pueden ser:

- Liberación cumplida de entregables
- Reducción de costos
- Logros o resultados significativos
- Uso de mejores prácticas
- Valores agregados
- Satisfacción demostrada de clientes y usuarios
- Calidad del software

Por otro lado, y dependiendo del nivel de riesgo del proyecto, es importante que se precisen las sanciones económicas en las que tanto la organización contratante como el proveedor podrían incurrir en caso de incumplimiento de las obligaciones contractuales. Dichas obligaciones contractuales tienen que ver con el incumplimiento injustificado de requerimientos o con retrasos

negligentes en las tareas que el proyecto define . Las sanciones económicas son útiles para corregir situaciones graves o para prevenir que ciertas situaciones indeseables se presenten en el futuro; sin embargo, se debe tener cuidado de no abusar de ellas y de aplicarlas solo como último recurso.

#### 5.3.2.6.3. Establecer las Obligaciones Contractuales

Dependiendo de las características particulares de cada proyecto y del tipo de contrato a ser aplicado, son muchas las consideraciones que ha nivel contractual deben guardar las partes implicadas. A continuación se sintetizan dichas obligaciones:

##### 5.3.2.6.3.1. Obligaciones por Parte del Proveedor

###### 5.3.2.6.3.1.1. *Cumplir con el Objeto de Contrato*

El objeto de contrato es la razón por la cual la organización contratante y el proveedor establecen una relación comercial. La organización contratante requiere de un servicio, un producto o una combinación de ambos, y paga al proveedor por la prestación de dicho servicio o por la adquisición o desarrollo del producto requerido. El proveedor obtiene un beneficio claro de esta relación, ya sea representado en dinero o ya sea representado en especies. Bajo este contexto, es de esperar que el proveedor cumpla con el objeto de negocio por el cual fue o será retribuido.

###### 5.3.2.6.3.1.2. *Realizar Estimados de Tiempo y Costos*

Bajo el contexto del desarrollo delegado de software (y de muchos proyectos similares), los costos de un contrato son calculados con base en el tiempo,

recursos y esfuerzo invertidos por el proveedor en el logro o cumplimiento del objeto contratado. Es obligación del proveedor realizar estimativos del tiempo y del esfuerzo requerido en cada una de las fases definidas para el proyecto.

#### *5.3.2.6.3.1.3. Monitorear y Reportar el Estado del Proyecto*

Es responsabilidad del proveedor monitorear el estado del proyecto en cada fase, calcular los indicadores previamente definidos en acuerdo con la organización contratante y mantener a esta última informada acerca del cumplimiento de los tiempos y costos pactados.

#### *5.3.2.6.3.1.4. Mantener el Valor del Contrato*

El proveedor tiene por obligación que mantener el precio pactado con la organización contratante por el objeto de negocio mientras dure el contrato. Dado que el proveedor es remunerado por el servicio, adquisición o desarrollo del producto requerido este tiene como responsabilidad mantener el valor del contrato acordado con el cliente. En casos muy excepcionales, como cuando la organización contratante incumple ciertas responsabilidades, el proveedor puede ser eximido de este compromiso. Como elemento implícito, se tiene que el proveedor es responsable por realizar una buena planificación ya que cualquier sobre costo generado por desfases o estimaciones de tiempos o costos incorrectas, es imputable a este.

#### *5.3.2.6.3.1.5. Propender por la Calidad de los Productos y Servicios Contratados*

En cuanto a productos software se refiere, la calidad tiene que ver con atributos intrínsecos muchas veces poco tangibles como lo son la funcionalidad, la usabilidad, la confiabilidad, el desempeño, la portabilidad y la mantenibilidad

[ISO91]. El proveedor tiene como responsabilidad asegurar la incorporación de dichos atributos en el producto software que le a sido delegado en la proporción que se haya definido con la organización contratante.

#### *5.3.2.6.3.1.6. Repudiar Presiones mal Intencionadas*

El proveedor tiene como responsabilidad informar a la organización contratante y a las autoridades que correspondan, acerca de cualquier presión, intimidación o chantaje que le sea realizado por parte de cualquier actor fuera de la ley para omitir o realizar algún acto o hecho mal intencionado.

#### *5.3.2.6.3.1.7. Responder por la Tercerización del Trabajo Asignado*

El proveedor esta en plena potestad de subcontratar parte o la totalidad del trabajo que le ha sido delegado por la organización contratante (si no se especifica lo contrario). En este caso, el proveedor esta obligado a responder ante el cliente por los actos o hechos que sean cometidos por los subcontratistas que se encuentren bajo su control. Implícitamente se define que la organización contratante se entenderá única y exclusivamente con el proveedor para efectos de asignación, control y cumplimiento de responsabilidades.

#### *5.3.2.6.3.1.8. Confidencialidad*

Por la naturaleza del proyecto, es muy probable que el proveedor tenga acceso y manipule mucha de la información privilegiada de la organización contratante. Además de las claras consideraciones éticas y morales allí implicadas, es necesario que se obligue contractualmente al proveedor a no divulgar la información que le ha sido confiada.

#### *5.3.2.6.3.1.9. Utilización de los Recursos*

Usualmente, en un proyecto de desarrollo delegado de software la organización contratante pone a disposición del proveedor una serie de recursos tales como servidores, software, equipos, documentación y similares. Es necesario que estos bienes sean guardados contractualmente para asegurar que se les da un buen uso, es decir, que están siendo empleados en actividades 100% relativas al proyecto. En adición, cualquier acto del proveedor o de sus subcontratistas que vaya en detrimento o que afecte el buen estado de los bienes entregados, será imputable al proveedor.

#### *5.3.2.6.3.2. Obligaciones por Parte de la Organización Contratante*

##### *5.3.2.6.3.2.1. Definir los Entregables del Proyecto*

La organización contratante tiene como responsabilidad asegurarse que desde el inicio del proyecto se definan los entregables que serán producidos en el proceso delegado de desarrollo de software. Esta responsabilidad se fundamenta en el argumento intrínseco que dicta que solo la organización contratante es consciente de sus propias necesidades y, consecuentemente, debe ser capaz de definir las. De igual modo, el cliente debe especificar la forma en que los productos serán entregados (liberaciones parciales o liberaciones completas) y la fecha de liberación de los mismos. La lista de entregables debe ser estrictamente definida en el documento de contrato, para que el cliente quede en potestad de exigirle al proveedor su cumplimiento.

#### *5.3.2.6.3.2.2. Hacer Seguimiento del Estado del Proyecto*

La organización contratante tiene como responsabilidad hacer seguimiento periódico del estado del proyecto. A través de algún mecanismo, como la definición de indicadores, el cliente debe verificar que se están cumpliendo los compromisos adquiridos por el proveedor en cuanto a costos y a plazos. Con el fin de prevenir conflictos con el proveedor, los mecanismos o estrategias que se empleen para realizar estas revisiones deben ser concertados y especificados contractualmente.

#### *5.3.2.6.3.2.3. Responsabilizarse por los Cambios Aprobados al Proyecto*

Es muy común que en los proyectos que tienen que ver con desarrollo de software, se presenten cambios sobre los requerimientos inicialmente planteados. Como ya se ha mencionado, es importante que los cambios que se puedan generar sobre los requerimientos del proyecto, máxime cuando estos implican modificaciones en tiempo, alcance o costos, sean gestionados apropiadamente bajo algún mecanismo previamente establecido. Si los cambios solicitados cursan dicho procedimiento y, si son aprobados, la organización contratante esta en la obligación de asumir los retrasos y sobrecostos ocasionados.

#### *5.3.2.6.3.2.4. Mantener las Condiciones Técnicas y Financieras Pactadas*

Así como el proveedor se encuentra obligado a mantener los acuerdos realizados en cuanto al valor del servicio o producto a desarrollar, la organización contratante también debe de respetar y mantener dichos acuerdos. En los contratos de desarrollo de software, es usual que el cliente suministre al proveedor recursos financieros y técnicos (servidores, equipos, materiales y similares) requeridos para que este último adelante

apropiadamente las labores que le han sido encomendadas. Si la provisión de dichos recursos es el fruto de compromisos contractuales adquiridos por la organización contratante, esta última está obligada a respetar dichos compromisos. En caso de incumplimiento, el éxito o fracaso del proyecto será atribuible al cliente.

#### *5.3.2.6.3.2.5. Identificación de Riesgos*

En cualquier proyecto de desarrollo de software, la identificación de riesgos es uno de los factores críticos de éxito más relevantes. Lo anterior se encuentra respaldado en el hecho que la gran mayoría de estándares y metodologías modernas o, al menos las más reconocidas, hacen grandes esfuerzos en torno al desarrollo de actividades que propenden por la correcta gestión de este tema. Dado que nadie mejor que el mismo cliente para reconocer las debilidades y fortalezas relativas a los activos de procesos, y a los recursos materiales, financieros y humanos propios, este es responsable de ayudar al proveedor a identificar los riesgos a los que se ve expuesto el proyecto. De igual modo, la organización contratante debe obligarse a participar activamente en la identificación y elaboración de planes de mitigación y de contingencia.

#### *5.3.2.6.3.2.6. Participar Activamente en el Proyecto*

El hecho que la organización contratante haya delegado en el proveedor las labores relativas al desarrollo del producto software, no la exime de participar en las actividades que el proyecto define. Esta participación se formaliza contractualmente con el claro establecimiento de los roles, actividades y responsabilidades a cargo del cliente. Debe considerarse que el recurso humano es un factor crítico para el éxito del proyecto y, en este sentido, el cliente debe asegurarse que el personal puesto a disposición del mismo cuente con el suficiente tiempo para cumplir con los roles asignados y para desarrollar

adecuadamente las tareas encomendadas. Las especificaciones contractuales en torno a roles y responsabilidades debe ser respetadas.

#### *5.3.2.6.3.2.7. Asignación de un Interventor*

Aunque no es estrictamente obligatorio, establecer la figura de un interventor es una práctica muy utilizada en proyectos extensos, costosos o de alto riesgo. El interventor es el encargado de velar porque se cumplan los compromisos pactados y porque se satisfagan los intereses del cliente; no obstante, la naturaleza de su labor también propende por la protección del proveedor y, en este sentido, dicha figura es beneficiosa para todas las partes involucradas en el proyecto. A continuación se nombran los objetivos más importantes de una interventoría:

- Verificar la ejecución y el cumplimiento de las actividades asignadas a los proveedores
- Propender por el cumplimiento de los objetivos de la contratación
- Vigilar la correcta ejecución del objeto contratado
- Proteger los derechos de la organización contratante, del proveedor y de otros terceros que puedan verse afectados por la ejecución del contrato
- Garantizar la neutralidad en el manejo de los conflictos que se puedan presentar entre las partes
- Asesorar a las partes en la elaboración, interpretación y seguimiento del contrato

Cabe añadir que los costos generados por la interventoría deben ser asumidos por la organización contratante con la salvedad, claro esta, que lo anterior no implica que la figura del interventor solo deberá propender por los intereses del cliente; una interventoría debe ir en pro del beneficio de todas las partes que se encuentren involucradas en el proyecto.

#### 5.3.2.6.4. Incorporación de Requerimientos y de Criterios de Aceptación

Como se expondrá posteriormente, los requerimientos expresan de manera formal los intereses, necesidades y expectativas de la organización contratante, la cual se encuentra representada por clientes y usuarios. Los requerimientos son uno de los artefactos (entregables) más relevantes en el proceso de adquisición, ya que prácticamente justifican la realización del proyecto, dirigen casi todas las fases de su ciclo de vida y son en si el objeto de contrato. El proveedor tiene como función principal satisfacer los requerimientos establecidos, para lo cual el cliente debe asegurarse de incorporarlos en el documento de contrato. De manera conjunta, la organización contratante debe establecer y añadir al contrato los criterios bajo los cuales el producto software a ser desarrollado será aceptado. Dichos criterios, denominados criterios de aceptación, son la base bajo la cual se determina si el producto software desarrollado por el proveedor satisface los requerimientos establecidos y, más importante aún, se convierten en el instrumento contractual que más claramente ampara los intereses del cliente. Tanto el tema de los requerimientos como el tema de los criterios de aceptación hacen parte integral de la propuesta desarrollada en este trabajo y serán descritos con mayor detalle en las secciones subsiguientes. Por último, en la tabla 5.24 se propone una lista de chequeo que identifica los aspectos mas relevantes que deben considerarse a la hora de elaborar un documento de contrato bajo las circunstancias impuestas por la tercerización de un proyecto de desarrollo de software:

<b>Generalidades</b>	
Establecer el propósito del contrato.	<input checked="" type="checkbox"/>
Identificación de las partes: nombre de la empresa, nit, dirección, teléfono, representante legal, etc.	<input checked="" type="checkbox"/>
Definir la terminología a utilizar.	<input checked="" type="checkbox"/>
Establecer el alcance del proyecto haciendo énfasis en los requerimientos del producto software, documentación a desarrollar, soporte, instalación y capacitación.	<input checked="" type="checkbox"/>

Tabla 5.24. Lista de chequeo para contratos de desarrollo de software.

<b>Requerimientos y Criterios de Aceptación</b>	
Especificación de requisitos del producto.	<input checked="" type="checkbox"/>
Establecer los procedimientos y criterios de aceptación del producto como: pruebas de aceptación completadas con un resultado satisfactorio, instalación completa, capacitación a usuarios completa, todos los entregables recibidos, corrección de errores encontrados en el producto en un periodo de tiempo predefinido después de la liberación del mismo.	<input checked="" type="checkbox"/>
Descripción de los criterios de calidad con los cuales el proveedor será medido.	<input checked="" type="checkbox"/>
Describir que constituye un desempeño óptimo del proveedor.	<input checked="" type="checkbox"/>
<b>Modificaciones</b>	
Describir las condiciones y procedimientos para realizar modificaciones o enmiendas sobre el alcance del proyecto.	<input checked="" type="checkbox"/>
Especificar quién está autorizado para realizar cambios en el contrato y responder preguntas acerca del mismo.	<input checked="" type="checkbox"/>
Especificar los procedimientos para el control de cambios sobre el contrato.	<input checked="" type="checkbox"/>
<b>Pagos y Especificaciones Económicas</b>	
Describir las condiciones de los pagos a realizar, consideraciones y cantidades.	<input checked="" type="checkbox"/>
Determinar que costos son permisibles, gastos, cargos y su variación.	<input checked="" type="checkbox"/>
Determinar cuando los pagos estarán asociados a entregables o hitos del proyecto.	<input checked="" type="checkbox"/>
Detallar acerca de los temas relativos a impuestos, pagos retrasados e intereses.	<input checked="" type="checkbox"/>
Especificar el uso de incentivos para entregas tempranas, reducción de costos, resultados significativos y logros, y el uso de "mejores prácticas".	<input checked="" type="checkbox"/>
Especificar las sanciones a las que se ven expuestos tanto el cliente como el proveedor en caso de retrasos o no cumplimiento de ciertos requerimientos. ¿Habrá reducciones de pago?	<input checked="" type="checkbox"/>
Especificar los montos permisibles por concepto de pago de derechos de autor y licencias.	<input checked="" type="checkbox"/>
<b>Naturaleza de las Relaciones</b>	
Especificar como serán manejados los temas concernientes a las relaciones públicas.	<input checked="" type="checkbox"/>
Especificar la naturaleza de la relación entre las partes en el contrato como lo puede ser la participación conjunta, una sociedad, relación empleado empleador, y similares.	<input checked="" type="checkbox"/>
<b>Vigencia del Contrato</b>	
Establecer el periodo de tiempo en el que el contrato se hará vigente y la fecha límite para realizar la entrega final. También es posible establecer fechas periódicas para realizar entregas parciales.	<input checked="" type="checkbox"/>

Tabla 5.24. Lista de chequeo para contratos de desarrollo de software<sup>33</sup>.

<sup>33</sup> Basado en: [SVE01], p. 51.

Obligaciones de la Partes, Derechos y Especificaciones Legales	
Establecer condiciones, provisiones y obligaciones concernientes a la terminación del contrato. El proveedor deberá liberar todos los materiales y entregables que están asociados con el contrato si este último este terminado y el cliente debe pagar por el trabajo desarrollado hasta ese momento.	☑
Establecer las responsabilidades de las partes para el mantenimiento de los compromisos pactados.	☑
Desarrollar las especificaciones relativas a los derechos de autor, patentes, secretos de marca, derechos de uso, licencias, y otros temas concernientes a la propiedad intelectual.	☑
Establecer las especificaciones legales concernientes a garantías, representación, incumplimiento, limitación de las responsabilidades, indemnizaciones, y similares.	☑
Describir las políticas relativas a la seguridad, no divulgación, acceso y confidencialidad de la información.	☑
Especificar los derechos y las obligaciones del proveedor para la asignación de sus deberes, repartición del trabajo y el uso de subcontratistas.	☑
Determinar cómo serán manejados los conflictos entre las partes y si se requerirá la intervención de algún tipo de entidad o intermediario.	☑
Describir los compromisos entre las partes involucradas y la división de responsabilidades, obligaciones y tareas.	☑
Especificar en qué medida y bajo qué condiciones la organización contratante estará involucrada en el proyecto.	☑
Especificar las condiciones de uso, rotación y reemplazo de personal clave.	☑
Especificar las condiciones acerca del uso de facilidades y equipos. También se deben considerar los temas relativos a las inversiones en este tipo de elementos ¿qué parte las debe realizar? ¿cuáles son los montos permisibles? ¿bajo que condiciones pueden realizarse?	☑
Especificar la estrategia a utilizar para monitorear y controlar el progreso del proyecto: estructura de desglose de trabajo, iteraciones o ciclos cortos, hitos, etc.	☑
Especificar la rata permisible de cambio de requerimientos (aplica al caso de los contratos de costos fijos unitarios).	☑
Especificar los procedimientos para reportar incidentes y el progreso del proyecto. Determinar también las métricas a utilizar y cuán a menudo serán reportadas por el proveedor.	☑
Especificar cualquier exigencia sobre el equipo de desarrollo que delegue el proveedor, acerca de los métodos de desarrollo y de los procesos de aseguramiento de la calidad.	☑
Especificar cualquier exigencia acerca del contenido del plan de desarrollo y la fecha límite que tiene el proveedor para entregarlo.	☑
Especificar cualquier requerimiento y/o restricción acerca del uso de COTS (ver sección 5.2.3.3.4) o determinado hardware por parte del proveedor. ¿Qué debería usar el proveedor en términos de herramientas hardware y software?, ¿qué no debería usar?	☑
Determinar los derechos de inspección del cliente y los procedimientos para efectuar auditorías.	☑
Firmas de las partes.	☑

Tabla 5.24. Lista de chequeo para contratos de desarrollo de software.

### **5.3.2.7. Revisar de Manera Conjunta los Requerimientos y el Contrato**

Con el propósito de asegurar un mutuo entendimiento de las necesidades de negocio, objetivos y alcance del proyecto, es necesario que tanto la organización contratante como el proveedor y otros terceros que se vean afectados por el contrato, hagan una revisión conjunta y exhaustiva tanto de los requerimientos establecidos como de las especificaciones contractuales. A nivel contractual, los puntos críticos susceptibles de revisión son:

- Criterios y procedimientos de aceptación
- Manejo de cambios en los requisitos
- Manejo de problemas después de la liberación del producto
- Obligaciones contractuales del proveedor
- Obligaciones contractuales de la organización contratante
- Temas relacionados a la propiedad intelectual, garantías y acuerdos de confidencialidad

Para mitigar el riesgo que se presenten malos entendidos o conflictos entre las partes, es conveniente que estas revisiones se conduzcan con un nivel de formalidad alto. Para dicho propósito, debe acudirse a sesiones formales de revisión, levantamiento de actas de asistencia y actas de aceptación. En las reuniones de revisión, deben exponerse de manera clara y concisa las especificaciones contractuales y los requerimientos del proyecto, y se debe asegurar que todas las partes involucradas logran un buen entendimiento de estos elementos. Cualquier solicitud de cambio que surja de estas revisiones, debe ser juiciosamente analizada y discutida a la luz de los intereses propios y, en caso de aprobación, debe levantarse un acta y efectuar los cambios que correspondan en los documentos de contrato y de requerimientos. En cualquier caso, una reunión final de aceptación debe ser realizada.

## 5.4. Administración del Proyecto

El propósito del proceso de Administración del Proyecto es monitorear el desempeño del proveedor, obtener información acerca del estado del proyecto, resolver problemas, administrar los pagos, y tratar con otros temas contractuales.

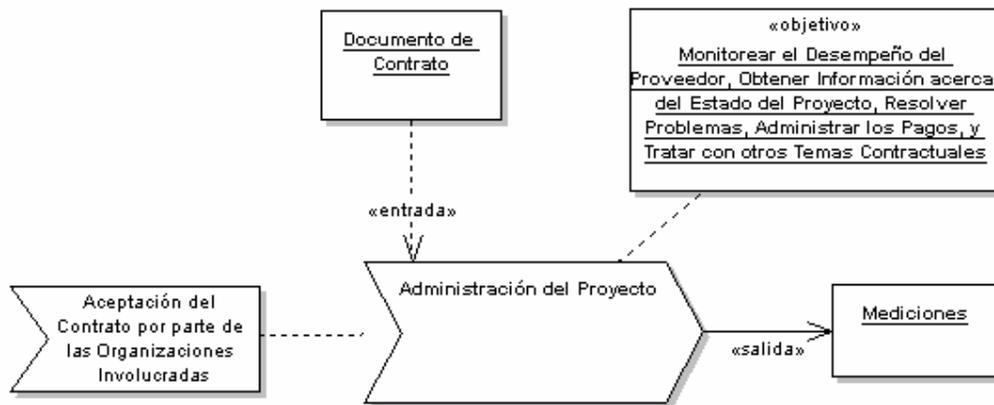


Figura 5.5. Diagrama de proceso de negocio del proceso Administración del Proyecto.

La figura 5.5 describe los elementos más importantes que intervienen en el proceso de Administración del Proyecto. El proceso inicia tan pronto las partes involucradas en el proyecto firman el contrato; de este último, extrae información concerniente a los compromisos adquiridos por el proveedor, plazos, pagos y similares, y que es de vital importancia para el logro de los objetivos que abajo se definen. Después de recorrer las actividades descritas en el proceso, se obtienen una serie de métricas que indican el estado del proyecto y que apoyan la toma oportuna de decisiones.

#### **5.4.1. Objetivos**

- Asegurar que los compromisos adquiridos por el proveedor están siendo cumplidos
- Revisar las métricas del proyecto y tomar las decisiones correspondientes
- Identificar de manera proactiva problemas e incidentes y escalarlos a quién corresponda
- Administrar los pagos a terceros y los cambios contractuales
- Asegurar que las tareas de finalización del proyecto se han llevado a cabo adecuadamente

#### **5.4.2. Actividades a Desarrollar**

- Velar por el Cumplimiento de las Obligaciones Contractuales del Proveedor
- Obtener Información Acerca del Estado del Proyecto
- Revisar el Estado del Presupuesto y del Plan del Proyecto
- Evaluar con Regularidad el Producto Software
- Reportar Oportunamente Mediciones y Problemas
- Establecer una Comunicación Regular y una Relación Colaborativa con el Proveedor
- Administrar Cambios Contractuales
- Administrar la Facturación
- Sellar el Contrato

#### **5.4.2.1. Velar por el Cumplimiento de las Obligaciones Contractuales del Proveedor**

La organización contratante debe velar constantemente porque el proveedor cumpla con sus obligaciones, tal cual como se pacto en el contrato. Por lo general, estas obligaciones se encuentran representadas en la forma de entregables o servicios a ser prestados. La organización contratante debe prestar especial atención a aquellos artefactos que tengan impacto directo sobre el producto software a ser desarrollado (componentes software, hardware, documentación técnica del producto y de usuario) y debe procurar que se entreguen en la fecha y con las especificaciones acordadas. Los insumos claves de esta actividad son el contrato y el cronograma del proyecto.

#### **5.4.2.2. Obtener Información Acerca del Estado del Proyecto**

Esta tarea implica la identificación y el establecimiento de los elementos del proyecto que deben ser monitoreados. Por lo general, para el monitoreo se realizan mediciones sobre presupuestos, progreso del proyecto, mitigación de riesgos, alcance, calidad de los entregables y similares. A continuación se exponen algunas de las métricas que podrían ser evaluadas:

- Hitos completados: básicamente se trata de comparar la fecha de terminación de cada uno de los hitos del proyecto contra lo que originalmente se planteó en el cronograma.
- Tasa de cambio de requisitos: número de requerimientos redefinidos / unidad de tiempo.
- Sobre tiempo del proveedor: se compara el tiempo adicional requerido por parte del proveedor para llevar a cabo las tareas asignadas respecto al tiempo previamente planificado para las mismas.

- Progreso de las pruebas: se pueden observar métricas como el número de casos de pruebas ejecutados contra lo planeado, pruebas fallidas vs. pruebas exitosas y similares.
- Reportes de problemas: número de problemas “abiertos” contra número de problemas “cerrados”.

#### **5.4.2.3. Revisar el Estado del Presupuesto y del Plan del Proyecto**

Con el fin de detectar desfases, de manera constante se debe revisar el presupuesto asignado para el proyecto respecto a lo gastado. Se deben tomar en cuenta no solo los pagos realizados al proveedor sino los posibles gastos internos atribuibles al desarrollo del producto en los que se haya podido incurrir. Conjuntamente, se debe realizar una revisión periódica del plan del proyecto para verificar que efectivamente este si se este llevando a cabo y que se estén cumpliendo los plazos establecidos. Situaciones anómalas como retrasos, pequeños retrasos acumulados y similares, deben ser prevenidas ya que tienen impacto directo tanto en tiempos como en costos.

#### **5.4.2.4. Evaluar con Regularidad el Producto Software**

Con el fin de anticipar incidentes en la entrega final del producto software, es importante tener en cuenta que los artefactos desarrollados en los distintos ciclos proyecto deben ser evaluados periódicamente. Con este propósito, se recomienda hacer referencia a las especificaciones descritas en el proceso de Evaluación del Producto el cual será tratado en la sección subsiguiente.

#### **5.4.2.5. Reportar Oportunamente Mediciones y Problemas**

Las métricas y mediciones que se levanten respecto al producto y a los procesos de adquisición establecidos deben ser transmitidas oportunamente a los participantes designados para su inclusión en el reporte de estado del proyecto. Esta información debe ser enviada a los patrocinadores del proyecto para su evaluación y para promover la toma de decisiones o acciones correctivas que puedan corresponder. Cualquier problema que afecte en tiempos, costos o calidad al producto software debe ser escalado de manera inmediata al participante que corresponda.

#### **5.4.2.6. Establecer una Comunicación Regular y una Relación Colaborativa con el Proveedor**

En aras del beneficio del proyecto de adquisición, se debe desarrollar una relación colaborativa con el proveedor. Es importante entender que de la efectividad y confianza que se cree en la relación con el proveedor depende en gran parte el éxito del proyecto y tanto la organización contratante como este último tienen responsabilidades en el mismo. En este sentido, se debe promover el trabajo en equipo, la resolución conjunta de problemas, la amabilidad, la puntualidad, el cumplimiento y, por sobre todas las cosas, debe prevalecer el respeto. Para favorecer este propósito, es recomendable formalizar y realizar reuniones periódicas con el proveedor donde se discutan aspectos del proyecto como costos, presupuestos, avances, plazos, inconvenientes, necesidades de capacitación y recursos, riesgos y estrategias de mitigación y, aún no menos importante, se obtenga una retroalimentación de parte y parte respecto a la satisfacción de expectativas.

#### **5.4.2.7. Administrar Cambios Contractuales**

Esta tarea tiene como propósito coordinar y comunicar a todos los participantes cualquier cambio contractual que se proponga realizar. Los cambios contractuales tienen gran impacto en los proyectos ya que por lo general pueden afectar costos, tiempos o alcance del producto. Por tal motivo, debe asegurarse que los cambios propuestos son analizados, documentados, acordados y notificados a las partes afectadas.

#### **5.4.2.8. Administrar la Facturación**

A pesar que muchas de las compañías que tercerizan la labor de desarrollo de software cuentan con personal especializado para gestionar todo lo que tiene que ver con pagos a proveedores, los participantes internos del proyecto no pueden desentenderse del todo de esta actividad. La razón se fundamenta en el hecho que lo facturado por los proveedores debe corresponderse con lo que en realidad estos han hecho dentro del proyecto. En este sentido, debe realizarse una verificación que involucre tanto a la organización contratante como al proveedor respecto a los pagos parciales realizados y a las actividades llevadas a cabo. Bajo el contexto de esta actividad, la organización contratante se refiere a los empleados de la misma que son participantes del proyecto de adquisición, que tienen el conocimiento del mismo y que son responsables de velar por el progreso del proyecto.

#### **5.4.2.9. Sellar el Contrato**

Cuando se realiza la liberación final del producto software y cuando los criterios de aceptación han sido satisfechos por parte del proveedor, el contrato debe ser completado y formalmente cerrado mediante un acta de finalización. Como es de esperar, cualquier asunto o problema que se encuentre pendiente debe

ser solucionado. El proveedor tiene la responsabilidad de terminar cualquier actividad o entregable que se encuentre incompleto y la organización contratante debe verificarlo. En este sentido, es muy recomendable congelar el último pago hasta que la entrega haya sido realizada a completa satisfacción. Finalmente, para llevar a cabo exitosamente esta tarea debe tenerse en cuenta las actividades descritas en el proceso de *Transición a Operaciones y Soporte* (ver numeral 5.6).

### **5.5. Evaluación del Producto**

El propósito del proceso de evaluación del producto es verificar que los entregables generados por el proveedor reúnen los requerimientos especificados y los objetivos de calidad. Lo anterior incluye la evaluación de entregas parciales o incrementales y la evaluación final, la cual es realizada mediante pruebas de aceptación. Básicamente, el propósito de este proceso puede resumirse en los siguientes ítems:

- Encontrar y documentar defectos de calidad en el producto
- Lograr una percepción aceptable en cuanto a la calidad del producto
- Validar que el producto cumpla con las especificaciones desarrolladas en la fase de diseño
- Validar que los requerimientos sean implementados apropiadamente

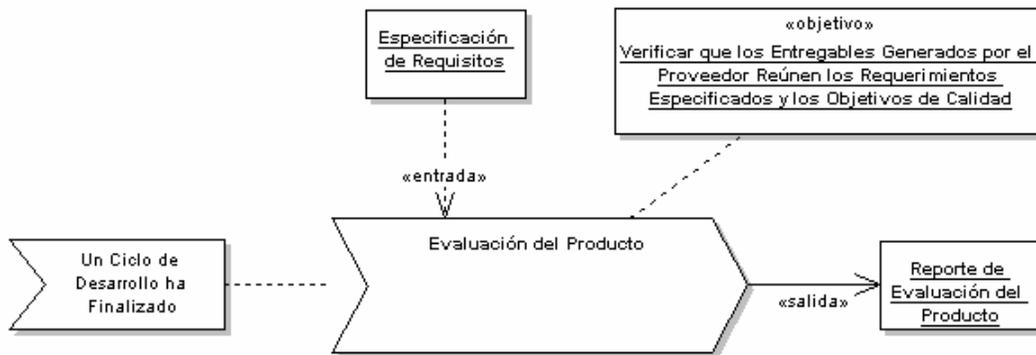


Figura 5.6. Diagrama de proceso de negocio del proceso Evaluación del Producto.

La figura 5.6 muestra los principales elementos que intervienen en el proceso de Evaluación del Producto. El proceso toma como insumo principal la especificación de requisitos e inicia cuando se detecta la finalización de un ciclo de desarrollo. Tras haberse ejecutado, el proceso genera un reporte de evaluación del producto en el cual se sintetiza los resultados de las pruebas efectuadas. Dichos resultados en conjunto con los criterios de aceptación definidos, determinan si el producto puede ser llevado o no a operaciones y soporte.

### 5.5.1. Objetivos

- Encontrar y documentar defectos en la calidad del producto
- Validar que el producto software trabaja según las especificaciones desarrolladas
- Validar que los requerimientos fueron implementados correctamente
- Lograr una percepción aceptable acerca de la calidad del producto

## **5.5.2. Actividades a Desarrollar**

- Establecer el Enfoque de las Pruebas
- Identificar los Artefactos a ser Evaluados
- Desarrollar las Especificaciones de las Pruebas
- Establecer Mecanismos de Pruebas para los Artefactos
- Definir los Requisitos de la Infraestructura de Pruebas
- Establecer y Desarrollar Evaluaciones Periódicas
- Analizar y Tomar Decisiones Frente a los Resultados de la Evaluación
- Concluir Pruebas de Aceptación

### **5.5.2.1. Establecer el Enfoque de las Pruebas**

Esta actividad tiene como propósito definir la estrategia de pruebas y las técnicas específicas a ser empleadas en el desarrollo de las mismas. Son muchas las técnicas que existen para evaluar un producto software, las cuales pueden ser conducidas durante todo el ciclo de vida del proyecto y durante las pruebas finales de aceptación. Es importante tener en cuenta que, bajo el contexto del desarrollo delegado de software, se puede hacer una distinción entre las pruebas que son aplicadas únicamente por el proveedor y las pruebas que deben ser aplicadas por la organización contratante. En este apartado se hace de una descripción de las pruebas donde la organización contratante toma parte. Las pruebas que son preliminarmente desarrolladas por el proveedor, como las pruebas unitarias y las pruebas de integración, no hacen parte del alcance de este trabajo.

#### 5.5.2.1.1. Pruebas Alfa

Las pruebas alfa son desarrolladas sobre versiones muy próximas a la de la liberación del producto software y son conducidas por la organización contratante en un ambiente controlado en el lugar de trabajo del proveedor y con los desarrolladores presentes [PRE00]. Generalmente, en las pruebas alfa los procedimientos para desarrollar las pruebas no se encuentran rigurosamente definidos; las funciones y tareas de negocio a ser exploradas son identificadas y documentadas, pero no existen casos de prueba específicos que se deban seguir. El cliente utiliza el software en la forma más natural posible, y los desarrolladores sólo observan y registran cualquier error o problema de uso.

#### 5.5.2.1.2. Pruebas Beta

Las pruebas beta son desarrolladas sobre versiones muy próximas a la de la liberación del producto software y son conducidas por los usuarios finales en uno o más sitios de la organización contratante [PRE00]. La principal diferencia entre las pruebas alfa y las pruebas beta, es que estas últimas son desarrolladas en el sitio real donde será destinado el software como producto final bajo los parámetros de configuración que requiere el ambiente de producción. Normalmente el proveedor no se hace presente durante este tipo de pruebas, lo que conlleva a que el producto trabaje bajo un ambiente que no puede ser controlado por el equipo de desarrollo.

#### 5.5.2.1.3. Pruebas Funcionales

Las pruebas funcionales se utilizan para demostrar que el producto software reúne y satisface los requisitos funcionales definidos, en otras palabras, confirman que el producto hace lo que los usuarios esperan que haga. Algo

importante con las pruebas funcionales, es que estas se desarrollan bajo el contexto de la perspectiva de los usuarios finales, los cuales deben ser eximidos de los detalles técnicos subyacentes. Este tipo de pruebas pueden ser realizadas a través de la utilización de casos de uso, o a través del desarrollo de un conjunto de pruebas que apliquen a cada uno de los requisitos funcionales, es decir, el diseño de casos de prueba. Es importante tener en cuenta que para lograr una adecuada cobertura de los escenarios de aplicación, no solo se requiere el ingreso de entradas válidas al producto software, sino también el ingreso de entradas erróneas.

#### 5.5.2.1.4. Pruebas de Instalación

Las pruebas de instalación pueden ser definidas como cualquier tipo de pruebas que se adelanten por fuera del entorno de desarrollo [WII06]. El propósito de conducir este tipo de pruebas es asegurar que el producto software puede ser instalado correctamente y, generalmente, son desarrolladas sobre el sistema computacional donde se espera que el producto trabaje. Es bastante generalizada la idea referente a que la instalación de un producto software simplemente se limita a la ejecución de un asistente de instalación. Las pruebas de instalación se concentran en asegurar que el producto quede instalado como se requiere, sobre distintas configuraciones de hardware o software y bajo distintas condiciones (por ejemplo, con poco espacio en disco o bajo interrupciones de energía).

#### 5.5.2.1.5. Pruebas de Soporte

Este tipo de pruebas son desarrolladas para asegurar que las necesidades del equipo que soportará y mantendrá el producto están cubiertas. Como se expondrá en el proceso de *Transición a Operaciones y Soporte* (ver sección 5.6), la organización de soporte y mantenimiento requiere de una serie de

recursos para desempeñar de manera eficaz y eficiente las labores de soporte del producto. Estos recursos pueden estar representados en la forma de hardware, software, documentación técnica, herramientas y similares, y pueden ser determinantes a la hora de solventar eventualidades asociadas al producto o cuando se hace mantenimiento preventivo del mismo. Algunos de los parámetros que pueden ser evaluados son [SUT06]:

- Pruebas sobre el volumen de llamadas
- Gestión de las llamadas encoladas
- Procedimientos para el escalamiento de problemas asociados al producto
- Procedimientos para el seguimiento de problemas
- Procedimientos para el reporte de problemas
- Disponibilidad, facilidad de acceso y velocidad de la base de conocimiento
- Lista de tarifas asociadas al soporte de incidentes
- Contactos para la asesoría del personal de soporte
- Hardware y software para reproducir problemas

#### 5.5.2.1.6. Pruebas de Recuperación

Las pruebas de recuperación tienen como objetivo la verificación que el producto software desarrollado puede recuperarse después de haberse presentado un fallo. En el contexto de los sistemas computacionales, los fallos se refieren a la pérdida de datos o de su integridad debido a anomalías en el funcionamiento del hardware, software o redes de datos. Para desarrollar este tipo de pruebas, es necesario simular las condiciones del fallo y aplicar los procedimientos de recuperación para llevar las bases de datos, las aplicaciones o el sistema a un estado deseable. Para aquellos productos que por su naturaleza deben mantenerse siempre disponibles, las pruebas de

recuperación aseguran que cuando un fallo ocurre, los sistemas de respaldo alternativos responden por el producto afectado sin ninguna pérdida de datos o de transacciones. Las siguientes condiciones son consideradas fuentes comunes de fallos:

- Interrupción de la potencia en el cliente
- Interrupción de la potencia en el servidor
- Interrupción de la comunicación vía servidores de red
- Interrupción de la comunicación o pérdida de poder en dispositivos de almacenamiento de datos
- Procesos de filtrado de datos interrumpidos
- Procesos de sincronización de datos interrumpidos
- Punteros a bases de datos o campos clave no válidos
- Datos inválidos o corruptos en la base de datos

Finalmente, debe tenerse en cuenta que las pruebas de recuperación son una estrategia altamente intrusiva y los procedimientos de desconexión de cables muchas veces no son deseables ni factibles. En este sentido, debe considerarse la utilización de métodos alternativos como lo son las herramientas software de diagnóstico.

#### 5.5.2.1.7. Pruebas de Seguridad

Las pruebas de seguridad tienen como propósito asegurar que los mecanismos de protección embebidos en el producto realmente lo protegen de penetraciones indebidas. Estos mecanismos se encuentran desarrollados con base en dos conceptos: autenticación y autorización. La autenticación tiene que ver con la identificación del actor que pretende acceder al producto. La autorización se refiere al nivel de acceso que tiene dicho actor sobre el producto.

#### 5.5.2.1.8. Pruebas de Stress

Las pruebas de stress tienen como propósito la verificación del desempeño del producto bajo condiciones extremas. Básicamente, la estrategia consiste en demandar recursos computacionales como memoria principal o procesador en una cantidad, frecuencia o volumen anormales. También es posible analizar el desempeño del producto cuando la disponibilidad de estos recursos se ve menguada o cuando el número de usuarios que intentan operar el producto es excesivo.

#### 5.5.2.1.9. Pruebas de Desempeño

Las pruebas de desempeño son implementadas y ejecutadas con el fin de evaluar las características de desempeño del producto como lo son la coordinación y el consumo de recursos de las distintas operaciones realizadas, los flujos de ejecución, los tiempos de respuesta y la confiabilidad y los límites operacionales. Adicionalmente, este tipo de pruebas buscan poner a punto el producto, no solo mediante la optimización de las características de desempeño del mismo, sino también mediante la optimización de los elementos que conforman el entorno en el cual este trabajará. En este sentido, la verificación de cómo el producto maneja condiciones extremas de carga y stress es determinante. Cabe anotar que dependiendo de la naturaleza del producto, por ejemplo, si se está tratando con sistemas en tiempo real o distribuidos, este tipo de pruebas pueden llevarse con menor o mayor rigurosidad.

#### 5.5.2.1.10. Pruebas sobre las Interfases de Usuario

Las pruebas sobre las interfases de usuario son realizadas junto con los usuarios finales y tienen como propósito evaluar la percepción que estos tienen

frente a lo apropiado e inapropiado de dichas interfases. Características tales como la facilidad de uso, los tiempos de respuesta y la experiencia de usuario son elementos esenciales de esta evaluación. Cabe anotar, que una de las características de los productos software más novedosos, es que el diseño de las interfases guían al usuario de manera intuitiva en la operación del producto, lo cual conlleva a beneficios adicionales como la facilitación de las actividades de capacitación y la disminución de la demanda de soporte. Son muchos los atributos asociados a las interfases de usuario, y su selección y consideración para efectos de verificación depende única y exclusivamente de las necesidades, facilidades y experiencia que se le quiera brindar a los usuarios. Algunos ejemplos de estos atributos pueden ser:

- Poder de búsqueda y selección
- Facilidades de ordenamiento
- Facilidades para la navegación
- Administración adecuada de ventanas
- Manejo de sesiones de usuario (almacenamiento de información)
- Ayuda en línea
- Capacidad para deshacer cambios

#### **5.5.2.2. Identificar los Artefactos a ser Evaluados**

Esta actividad tiene como propósito identificar los elementos del sistema que requieren ser probados y que tienen una fuerte relación con los factores críticos de pruebas.

##### **5.5.2.2.1. Identificar Factores Críticos de Pruebas**

Antes de establecer los artefactos que van a ser evaluados, es necesario identificar y entender todos aquellos elementos inmersos en el proyecto que

son clave, y que justifican y estimulan el proceso de evaluación que se pretende desarrollar. Estos elementos, denominados factores críticos de pruebas <sup>34</sup>, pueden definirse como cualquier proceso, actividad, tarea, entregable o evento del producto software, que estimule a emprender el esfuerzo de pruebas o que impulse a los integrantes del equipo de pruebas a desarrollarlas. Los factores críticos de pruebas ayudan a identificar y a hacer visibles las cosas que motivarán a los integrantes del equipo de pruebas a evaluar los aspectos clave del producto software. Generalmente, dichos aspectos toman la forma de riesgos específicos de calidad que ayudan a determinar la importancia relativa a cada factor crítico, y pueden exponer otros factores críticos que no hayan sido considerados. Los factores críticos de pruebas pueden provenir de fuentes tales como entregables, eventos ante los cuales el producto software tenga que responder, actividades que se tengan que llevar a cabo dentro del proceso de desarrollo o la total ausencia de estos elementos. Ejemplos de estas fuentes pueden ser los modelos UML (Lenguaje Unificado de Modelamiento, por su nombre en español)<sup>35</sup>, casos de uso, las especificaciones de riesgos del proyecto, solicitudes de cambio, especificaciones de requisitos y similares.

#### 5.5.2.2.2. Identificar Elementos Dinámicos y Eventos del Producto

El propósito de esta tarea es lograr un entendimiento de los aspectos dinámicos y de tiempo de ejecución del producto desarrollado para identificar cuáles de ellos son relevantes o asocian artefactos críticos que deben ser evaluados. Para identificar estos aspectos, puede recurrirse a la especificación de requisitos o a artefactos elaborados por el proveedor en la fase de diseño como lo son los casos de uso; modelos de diseño, implementación y despliegue; clases de control, procesos, eventos y similares. Una buena estrategia que puede ayudar a guiar esta búsqueda, consiste en identificar

---

<sup>34</sup> [RUP03], disciplina de pruebas.

<sup>35</sup> Ver [JBR99] para mas ampliación acerca del tema de UML.

todos aquellos artefactos que representen flujos de trabajo, o que describan de algún modo secuencias de eventos o procesos de negocio. Si se está utilizando UML como notación, estos elementos pueden verse representados por componentes marcados con los estereotipos de control o proceso, o por asociaciones de realización entre componentes.

#### 5.5.2.2.3. Identificar los Límites del Sistema y sus Interfases

Con el fin de entender el rol que el producto software juega como proveedor de servicios y sus dependencias como cliente, es necesario que se clarifiquen los límites del sistema y sus interfases. Las interfases son importantes porque definen los límites del sistema y ayudan a entender la relación del producto software con actores externos los cuales, básicamente, pueden ser de dos tipos: otros sistemas y usuarios finales. Los actores que son representados por otros sistemas establecen una relación con el producto software ya sea como clientes, como proveedores de servicios o como ambos. Esta información es bastante valiosa ya que de ella se desprenden una serie de necesidades, expectativas o requerimientos que dichos actores demandan del producto software en evaluación y, con base en ello, se pueden establecer las validaciones necesarias en términos de estas interfases y de la infraestructura de pruebas requerida para soportarlas. Al igual que en la identificación de los elementos dinámicos y de los eventos del producto, es recomendable recurrir a los modelos elaborados en la fase de diseño y/o a los modelos de implementación y de despliegue, para levantar la información que se precise.

#### 5.5.2.3. **Desarrollar las Especificaciones de las Pruebas**

Esta actividad tiene como objetivo especificar de manera formal las pruebas que han de realizarse sobre los artefactos objetivo y los procedimientos que deben seguirse para dicho propósito.

#### 5.5.2.3.1. Identificar Casos de Prueba

Los casos de prueba son una estrategia fundamental en el proceso de evaluación del producto ya que mediante ellos puede lograrse una cobertura tan amplia como se decida y como sea viable, de los aspectos funcionales y no funcionales del producto software. La palabra *particular* tiene una connotación especialmente importante dentro del contexto de esta actividad. Cada artefacto posee una serie de características que lo identifican y que lo hacen único en el proceso de evaluación. Dependiendo de los requerimientos funcionales y no funcionales que se hayan definido, un conjunto de dichas características toman especial importancia mientras que el resto quedan relegadas. Las características que se hayan considerado como importantes, adquieren la forma de aspectos específicos que se requieren evaluar respecto a ese artefacto. Dichos aspectos pueden ser considerados como escenarios de aplicación claves donde el artefacto en cuestión juega un papel protagónico y sobre los cuales deben desarrollarse validaciones, es decir, casos de prueba.

Para identificar casos de prueba, al igual que en otras actividades de evaluación, puede recurrirse a los modelos desarrollados en las fases de análisis y de diseño. Sin embargo, existen artefactos que son fuentes “cuasi explícitas” de casos de pruebas. El primero de ellos es el modelo de casos de uso. Los casos de uso definen por si solos escenarios de aplicación y, por tanto, casos de prueba. De igual modo, las asociaciones de generalización, extensión e inclusión que se establecen entre ellos, pueden representar con gran certeza escenarios de aplicación. Otra fuente importante de casos de prueba es la misma especificación de requisitos. En una especificación de requisitos se definen una serie de restricciones que deben imponerse sobre ciertos elementos del sistema. Estas restricciones generalmente representan lógica de negocio y son aspectos particulares de la aplicación que deben ser evaluados a través de casos de pruebas. Por último, y si se están utilizando guiones

gráficos<sup>36</sup> como técnica para la elicitación de requisitos, estas historietas son en si mismas escenarios de aplicación a los cuales puede asociárseles casos de prueba.

#### 5.5.2.3.2. Especificar los Casos de Prueba

Esta tarea tiene como propósito realizar la especificación formal de los casos de pruebas asociados al aspecto particular que se desea evaluar del artefacto objetivo. Adicionalmente, esta labor ayuda a llevar un control efectivo de los casos de prueba definidos, pudiendo identificar en todo momento el estado de los mismos y el artefacto al cual se pretende evaluar. Según la explicación desarrollada en el punto anterior, para un artefacto en particular pueden definirse múltiples pruebas ya que cada prueba representa un escenario de aplicación particular para dicho artefacto. Lo anterior implica que debe manejarse algún tipo de llave que permita diferenciar entre los distintos casos de prueba asociados a un mismo artefacto. A continuación se listan los elementos más importantes que hacen parte de la estructura de los casos de prueba:

<b>Para cada caso de prueba se debe incluir:</b>	
Identificador y nombre del caso de prueba.	<input checked="" type="checkbox"/>
Descripción, objetivo del caso de prueba	<input checked="" type="checkbox"/>
Requerimiento, caso de uso al que asocia (especifica lo que se quiere examinar).	<input checked="" type="checkbox"/>
Datos de entrada	<input checked="" type="checkbox"/>
De adonde provienen los datos (p.e., entrada manual, dispositivo de campo, entrada simulada, etc.).	<input checked="" type="checkbox"/>
¿Cuánto tiempo se prolongará la prueba? (p.e., recolectar datos generados dentro de un bucle por una hora).	<input checked="" type="checkbox"/>
Valor(es) esperados de salida / resultados esperados	<input checked="" type="checkbox"/>
Criterios de aceptación (determinan si la prueba pasa o falla).	<input checked="" type="checkbox"/>
Procedimiento de pruebas al que asocia.	<input checked="" type="checkbox"/>

Tabla 5.25. Lista de chequeo para los casos de prueba<sup>37</sup>.

<sup>36</sup> Ver [GLM06] para profundizar más en el tema de los guiones gráficos.

### 5.5.2.3.3. Desarrollar Procedimientos de Pruebas

El propósito de esta tarea es desarrollar y asociar a cada caso de prueba un procedimiento de prueba. Un procedimiento de prueba es una guía que describe paso a paso las acciones que deben adelantarse para ejecutar con éxito un caso de prueba. Cada caso de prueba es desarrollado a través de un procedimiento específico de prueba que indica de manera explícita y concreta lo que se debe realizar para obtener resultados del caso de prueba. Los procedimientos de pruebas son importantes porque evitan la improvisación y la consecuente ineficiencia en la ejecución de los casos de pruebas.

La tabla 5.26 describe los elementos que componen un procedimiento de prueba:

<b>Para cada procedimiento de prueba se debe incluir:</b>	
Actividades previas requeridas para habilitar la prueba (p.e., apagar o prender ciertos equipos, o cargar cierta pieza de software en un servidor).	☑
Referencia al requerimiento o caso de prueba para el cual fue construido.	☑
Una descripción detallada, paso a paso, de las acciones requeridas para llevar a cabo la prueba (este es el procedimiento de prueba como tal).	☑
Los procedimientos utilizados para reducir y analizar los datos (p.e., explícitamente establecer las ecuaciones, las fórmulas estadísticas, las técnicas de promediación, etc.).	☑
Equipos requeridos para correr el procedimiento y para analizar datos.	☑
Especificar cualquier dispositivo de campo que se requiera (p.e., una impresora si algún paso del procedimiento involucra una impresión).	☑
Especificar cualquier interfase que se requiera con otros sistemas (p.e., un sistema legado).	☑

Tabla 5.26. Lista de chequeo de los procedimientos de prueba<sup>38</sup>.

<sup>37</sup> Basado en: [SAA98], v. 2, p. 15-12.

<sup>38</sup> Basado en: [SAA98], v. 2, p. 15-11.

#### **5.5.2.4. Establecer Mecanismos de Pruebas para los Artefactos**

El objetivo principal de esta actividad es establecer los mecanismos de pruebas mas apropiados para cada uno de los artefactos seleccionados para evaluación. Después de haber definido dichos artefactos, es necesario identificar las técnicas, enfoques o mecanismos de pruebas que mejor puedan soportar las evaluaciones a realizar y que mejor apoyen los objetivos de las mismas; en palabras simples, a cada artefacto debe asociársele un enfoque o técnica de pruebas. Con este fin, debe analizarse cuán cerca están los mecanismos de prueba seleccionados de proveer una solución de pruebas completa y como pueden ser adaptados para que se ajusten mejor a los requerimientos de evaluación planteados para cada uno de los artefactos. Si no se encuentran mecanismos de pruebas apropiados o si su adaptación requiere de un gran esfuerzo, deben definirse nuevos mecanismos tratando de lograr un equilibrio entre la especificidad de los mismos y la posibilidad de reutilización. Los mecanismos o enfoques de pruebas que mejor aplican al caso del desarrollo delegado de software son descritos con mayor detalle en la actividad Establecer el Enfoque de las Pruebas (ver sección 5.5.2.1).

#### **5.5.2.5. Definir los Requisitos de la Infraestructura de Pruebas**

El propósito de esta actividad es definir los requisitos que la infraestructura de pruebas debe reunir para soportar el desarrollo de las actividades de evaluación. Básicamente se trata de una actividad de apoyo donde se definen e implementan requerimientos en torno a la infraestructura de pruebas que permitirá al equipo que desarrollará el proceso de evaluación desempeñar sus labores de una manera mas eficiente. Estos requerimientos pueden hacer referencia a servicios técnicos tales como la administración de sistemas hardware y software, las telecomunicaciones, la creación y reproducción de documentos, la creación de copias de respaldo y similares. Algunos requisitos básicos que podrían definirse son:

- La creación de cuentas de usuario
- La instalación del software requerido para diseñar y desarrollar las pruebas
- La resolución de problemas potenciales de red
- La adecuación de sistemas hardware
- La adecuación o desarrollo de documentación técnica
- La configuración de sistemas software

Algunas de las tareas que bajo esta actividad deben desarrollarse son:

- Identificar requerimientos específicos de infraestructura para cada mecanismo de pruebas
- Definir el inventario de hardware y software requeridos para desarrollar las pruebas
- Definir y aplicar las configuraciones necesarias para el hardware y software requeridos
- Definir los procedimientos a seguir para mantener el ambiente de pruebas en buenas condiciones

La última tarea define un tema importante en torno a la infraestructura de pruebas que es el que tiene que ver con los ambientes de pruebas. Los ambientes de pruebas definen una instancia específica de configuración de hardware y software que se establece con el propósito de desarrollar pruebas bajo condiciones controladas y conocidas<sup>39</sup>; en otras palabras, habilitan el desarrollo de mecanismos de pruebas tales como las pruebas alfa. Como se mencionó anteriormente, se debe tener en cuenta que para que los ambientes de pruebas sean realmente provechosos, estos deben reflejar con la mayor fidelidad posible las configuraciones y variables contenidas en los ambientes de

---

<sup>39</sup> [RUP03], disciplina de entorno.

producción. Lo anterior implica el desarrollo de labores de infraestructura adicionales como lo son la creación de objetos (por ejemplo, bases de datos, tablas, y/o procedimientos almacenados) y el cargue masivo de datos.

#### **5.5.2.6. Establecer y Desarrollar Evaluaciones Periódicas**

Con el fin de minimizar el riesgo que el producto software no reúna las expectativas del usuario final, es imprescindible establecer evaluaciones periódicas y desarrollar planes para las mismas. A continuación se describen algunas de las tareas que deben desarrollarse para cumplir con este propósito.

##### **5.5.2.6.1. Determinar los Entregables del Próximo Ciclo**

Esta tarea tiene como propósito facilitar las labores de planificación de las evaluaciones periódicas recurriendo a la identificación oportuna de los artefactos que serán entregados en el próximo hito, fase o etapa del proyecto. Tomando como recurso primordial el plan del proyecto, deben identificarse los componentes software que serán desarrollados por el proveedor en la próxima fase o etapa del proyecto. De igual modo, deben determinarse los artefactos asociados a dichos componentes o que deben ser elaborados durante el próximo ciclo y que también deben ser tomados en consideración para la evaluación venidera. Estos artefactos pueden referirse a documentación técnica, eventos asociados al producto, modelos UML, actividades y similares.

De manera análoga a la identificación de los componentes software a desarrollar en el próximo ciclo, en esta tarea también deben tomarse en consideración las solicitudes de cambio que se hayan realizado y las modificaciones en el alcance del proyecto. Es usual que las solicitudes de cambio realizadas hagan referencia o afecten a componentes software desarrollados en etapas anteriores del proyecto y, por lo tanto, se hace

necesario reevaluar tanto esos componentes como los componentes software con los cuales interactúen o tengan alguna relación. De manera consecuente, las modificaciones en el alcance que hayan sido aprobadas y que definan adiciones al producto software inicialmente especificado, también definen nuevos artefactos que deben ser tenidos en cuenta para las próximas evaluaciones.

#### 5.5.2.6.2. Identificar los Elementos del Sistema a ser Probados

Además de los componentes software a ser desarrollados en el próximo ciclo y de los artefactos asociados a ellos, es imprescindible identificar todos aquellos elementos que conforman el sistema para evaluar si se encuentran preparados para el despliegue de dichos componentes. Estos elementos pueden referirse a dispositivos hardware, controladores, componentes software de terceros, sistemas operativos, software de red y de comunicaciones y similares, o a las configuraciones a aplicar sobre dichos elementos. Se debe recordar que, bajo este contexto, cuando se habla de sistema se está hablando del producto software, el proceso o procesos de negocio que soporta, los usuarios, los eventos, las actividades y las interacciones entre ellos. Es importante también considerar los ambientes sobre los cuales estos elementos serán probados para que no se presenten inconsistencias en el paso a producción. En este sentido, es recomendable que el ambiente de pruebas simule lo más fielmente posible al ambiente de producción y, en la medida de lo posible, sean consideradas dentro de todas las variables que aplican a este último.

#### 5.5.2.6.3. Elaborar un Plan de Evaluación

El propósito de la elaboración de un plan de evaluación es definir los objetivos, el alcance, los artefactos a probar, el enfoque de pruebas a utilizar, los recursos requeridos y los entregables a desarrollar. Es importante tener en

cuenta que la cantidad y prolongación de las fases y etapas del proyecto, tienen una gran incidencia en la forma en que deben planificarse las evaluaciones que se desarrollaran durante el mismo, es decir, si las fases o etapas del proyecto son prolongadas habrá una brecha de tiempo mayor entre las evaluaciones que se establezcan; en contraposición, si las fases o etapas del proyecto son más cortas, las labores de evaluación se adelantarán con una frecuencia mayor. Por otro lado, entre más fases y etapas defina el proyecto más evaluaciones deberán realizarse si se quieren identificar de manera pronta posibles problemas e inconsistencias. Debe tenerse en cuenta que si el proyecto en ejecución define muchos riesgos o si la probabilidad que estos se vuelvan realidad es muy elevada, es recomendable establecer fases y etapas cortas para advertir de manera oportuna los problemas. A continuación se describen brevemente los elementos que debería incluir el plan de evaluación:

<b>Generalidades</b>	
Describir los objetivos y el alcance de las pruebas.	<input checked="" type="checkbox"/>
¿Qué cubrimiento tendrán las pruebas? Se elabora un listado de los casos de uso/requerimientos a evaluar.	<input checked="" type="checkbox"/>
<b>Las Organizaciones Involucradas y sus Respective Roles</b>	
¿Quién será responsable de conducir las pruebas?	<input checked="" type="checkbox"/>
¿Quién será responsable de registrar los datos?	<input checked="" type="checkbox"/>
¿Quién será responsable por analizar los datos y reportar los resultados?	<input checked="" type="checkbox"/>
<b>Lugar de las Pruebas</b>	
¿Dónde serán efectuadas las pruebas? ¿En las instalaciones del cliente? ¿En las instalaciones del proveedor? ¿En un lugar alternativo?	<input checked="" type="checkbox"/>
<b>Cronograma de Pruebas</b>	
Establecer un cronograma para las evaluaciones periódicas y para la evaluación final del producto.	<input checked="" type="checkbox"/>

Tabla 5.27. Lista de chequeo plan de evaluación<sup>40</sup>.

<sup>40</sup> Basado en: [SVE01], p. 41.

<b>Recursos Requeridos para Adelantar las Pruebas</b>	
Especificar los computadores donde serán llevados a cabo las pruebas; estos contendrán la copia del producto software o accederán a el a través de la red.	<input checked="" type="checkbox"/>
Especificar cualquier dispositivo de campo que se requiera (p.e., una impresora si se requieren hacer pruebas de impresión, o un sensor, etc.).	<input checked="" type="checkbox"/>
Especificar cualquier interfase que se requiera con otros sistemas (p.e., un sistema legado).	<input checked="" type="checkbox"/>
Especificar cualquier software especializado que se requiera para llevar a cabo las pruebas (p.e., software para simular el stress al cual será sometido el producto).	<input checked="" type="checkbox"/>
Especificar la documentación requerida para adelantar las pruebas (p.e., especificación de casos de prueba, procedimientos de prueba, etc.)	<input checked="" type="checkbox"/>
<b>Entorno de Pruebas</b>	
Hardware requerido y la configuración a aplicar.	<input checked="" type="checkbox"/>
Software requerido y la configuración a aplicar.	<input checked="" type="checkbox"/>
¿Qué datos se deben cargar en el ambiente de pruebas? ¿Dónde se encuentran estos datos?	<input checked="" type="checkbox"/>
Variables del dominio a introducir o simular en el ambiente de pruebas (p.e., la frecuencia con que un usuario realiza una operación sobre el sistema).	<input checked="" type="checkbox"/>
<b>Criterios de Aceptación</b>	
Debe especificarse las tasas aceptables para los fallos (p.e., pasar todas las pruebas catalogadas como críticas y el 80% de las restantes).	<input checked="" type="checkbox"/>
<b>¿Qué pasa cuando una prueba falla o no se lleva a cabo como se planeó?</b>	
Especificar como serán adelantadas las pruebas de regresión en caso que se necesiten.	<input checked="" type="checkbox"/>
<b>Lista de Pruebas a ser Corridas</b>	
Se asigna un identificador a cada prueba (p.e., Prueba 1A).	<input checked="" type="checkbox"/>
Se describe el propósito de cada una de las pruebas.	<input checked="" type="checkbox"/>
Se especifican los datos a ser registrados; estos son datos que se generan mientras se ejecuta el procedimiento de prueba de cada prueba.	<input checked="" type="checkbox"/>
Se especifican los criterios de éxito o falla de cada una de las pruebas (dependiendo del tipo de prueba a realizar, un caso de prueba puede ser mas adecuado para este fin).	<input checked="" type="checkbox"/>
<b>Trazabilidad</b>	
Para cada prueba se debe indicar el requerimiento(s) que examina (Nota. Se trata de mostrar que las pruebas están basadas en requerimientos; esto puede ser visto como una medida de protección para el proveedor, para evitar que de manera soterrada se agreguen requerimientos que no estaban contemplados en el alcance).	<input checked="" type="checkbox"/>
Para cada requerimiento se debe indicar la prueba(s) que lo examinará. (Nota. Se trata de indicar el cubrimiento que tendrán las pruebas; esto puede ser visto como una medida de protección para el cliente, dado que se garantiza que todos los requerimientos serán cubiertos con, al menos, una prueba).	<input checked="" type="checkbox"/>

Tabla 5.27. Lista de chequeo plan de evaluación.

### 5.5.2.7. Analizar y Tomar Decisiones Frente a los Resultados de la Evaluación

Después de haber corrido las pruebas especificadas, es necesario realizar un análisis minucioso de los resultados obtenidos. Básicamente, se trata de capturar los detalles generados en la ejecución de las pruebas, establecer la calidad percibida del producto, y tomar las acciones correctivas que sean necesarias. A continuación se describen algunas de las tareas que deben llevarse a cabo:

#### 5.5.2.7.1. Examinar los Incidentes y Fallos Presentados

Esta tarea tiene como objetivo fundamental la identificación de desviaciones significativas respecto a los resultados esperados de una prueba. Para dicho propósito, se toma como insumo principal los registros de pruebas generados durante la ejecución de las pruebas, y se examinan los datos allí contenidos para tratar de identificar inconsistencias.

La tabla 5.28 elabora una lista de chequeo para el contenido mínimo de un registro de pruebas. Debe recordarse, que los resultados esperados son especificados en la descripción de cada caso de prueba (ver tabla 5.25). Es primordial, que el desarrollo de esta tarea sea siempre orientado a la identificación de relaciones entre los incidentes presentados, tales como fallos repetitivos, incidentes duplicados, síntomas comunes y similares. Cada incidente debe ser analizado de la manera mas rigurosa posible, de manera tal que la causa raíz del problema pueda ser establecida.

Para cada registro de prueba incluir:	
Identificador/nombre de la prueba.	<input checked="" type="checkbox"/>
Nombre o identificador del procedimiento de prueba con que se asocia.	<input checked="" type="checkbox"/>

Tabla 5.28. Lista de chequeo para el registro de pruebas<sup>41</sup>.

<sup>41</sup> Basado en: [SAA98], v. 2, p. 15-13.

Fecha y hora en que la prueba empezó.	<input checked="" type="checkbox"/>
Fecha y hora en que la prueba terminó.	<input checked="" type="checkbox"/>
¿Quién llevó a cabo la prueba?	<input checked="" type="checkbox"/>
Especificar cualquier desviación respecto al procedimiento de prueba (p.e., el ejecutor de la prueba inadvertidamente se saltó un paso. Por lo tanto, un error se generó en el paso siguiente y fue corregido manualmente en plena ejecución del procedimiento).	<input checked="" type="checkbox"/>
Salidas generadas en la ejecución del procedimiento (p.e., registros insertados a una tabla de una base de datos; cuadro de diálogo de confirmación de una operación realizada sobre el sistema).	<input checked="" type="checkbox"/>

Tabla 5.28. Lista de chequeo para el registro de pruebas<sup>42</sup>.

#### 5.5.2.7.2. Valorar la Calidad del Producto

Con la experiencia obtenida en la tarea anterior, ahora es posible clarificar la calidad percibida del producto software o del artefacto que se tenga en cuestión. Esta valoración puede hacerse tan flexible o tan rigurosamente como se requiera, pero siempre debe estar orientada a la verificación del cubrimiento de las necesidades del negocio, de los usuarios y de los clientes. Con este fin, es recomendable recurrir a los requerimientos de calidad que hayan sido establecidos, y cotejarlos uno a uno contra el impacto generado por los incidentes y fallos encontrados. Si no han sido establecidos de manera formal este tipo de requerimientos, se puede acudir a cualquiera de los modelos de calidad existentes en la industria y basarse en ellos para extraer y ponderar aquellas características y criterios que a la luz del proyecto en curso se consideren importantes. Recuérdese que en el marco teórico de este trabajo se describió uno de estos modelos (el modelo de McCall), pero también es posible emplear cualquier otro que sea lo suficientemente reconocido, fácil de usar y efectivo (p.e., ISO 9126). Por último, y con el ánimo de simplificar esta labor, puede simplemente hacerse un recorrido por los aspectos buenos y malos encontrados en el producto, y con base en esta información y de acuerdo a las necesidades organizacionales tomar las decisiones que correspondan.

<sup>42</sup> Basado en: [SAA98], v. 2, p. 15-13.

### 5.5.2.7.3. Desarrollar el Reporte de Pruebas de Aceptación

Como paso previo a la terminación de este proceso, es importante que se desarrolle un reporte de pruebas de aceptación. Este reporte actuará como el soporte requerido para aceptar el producto o artefacto evaluado y dar vía libre al proceso de transición a operaciones y soporte del mismo. La tabla 5.29 especifica los ítems que como mínimo el reporte debe incluir:

<b>Información General</b>	
Periodo o rango de fechas en que las pruebas fueron desarrolladas.	<input checked="" type="checkbox"/>
Número de casos de prueba planeados para pruebas.	<input checked="" type="checkbox"/>
Número de casos de prueba corridos.	<input checked="" type="checkbox"/>
Número de casos de prueba que pasaron la prueba.	<input checked="" type="checkbox"/>
Número de casos de prueba fallidos.	<input checked="" type="checkbox"/>
<b>Aceptación del Producto</b>	
¿El producto software pasa las pruebas? En caso afirmativo, se debe especificar los pasos o recomendaciones a seguir (Nota. La aceptación o rechazo de un producto software se determina de acuerdo a los criterios establecidos en el plan de pruebas).	<input checked="" type="checkbox"/>
En caso que el producto haya sido rechazado deben especificarse la razones de esa decisión (esto también se determina de acuerdo a los criterios establecidos en el plan de pruebas).	<input checked="" type="checkbox"/>
<b>Lista de Resultados de Casos de Prueba</b>	
Identificador/nombre del caso de prueba.	<input checked="" type="checkbox"/>
Procedimiento de prueba utilizado para desarrollar el caso de prueba.	<input checked="" type="checkbox"/>
Cualquier desviación en la que se haya incurrido respecto al procedimiento de prueba.	<input checked="" type="checkbox"/>
Registros de prueba generados en la corrida.	<input checked="" type="checkbox"/>
Datos computados (p.e., si se hizo algún cálculo para verificar que los datos obtenidos correspondían con lo que se esperaba).	<input checked="" type="checkbox"/>
Indicar si la prueba pasó o falló (esto se establece de acuerdo a los criterios de aceptación documentados en la especificación de los casos de uso).	<input checked="" type="checkbox"/>

Tabla 5.29. Lista de chequeo del reporte de pruebas de aceptación<sup>43</sup>.

<sup>43</sup> Basado en: [SVE01], p. 55.

### 5.5.2.8. Concluir Pruebas de Aceptación

Como paso final, y formalismo necesario, el reporte de pruebas de aceptación es validado tanto por la organización contratante como el proveedor y, en caso de no haber ninguna objeción, se da vía libre a la transición del producto.

## 5.6. Transición a Operaciones y Soporte

El propósito del proceso de transición para el soporte es la minimización de los traumatismos generados por la transición de los entregables del proyecto a la organización de soporte y mantenimiento y la verificación que dichos entregables y otros servicios, como capacitación a los usuarios e instalación del producto software, han sido recibidos a satisfacción.

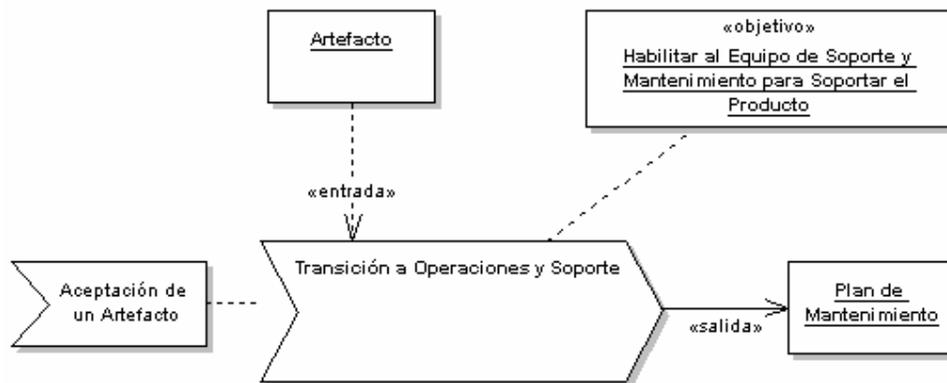


Figura 5.7. Diagrama de proceso de negocio del proceso Transición a Operaciones y Soporte.

La figura 5.7 describe los principales elementos del proceso de Transición para el Soporte y Operaciones. El proceso inicia cuando la aceptación de un artefacto ha sido recibida. Como ya se ha mencionado, los artefactos pueden referirse a componentes software, hardware, documentación técnica, manuales de usuario, dispositivos de campo, ejecutables, planes y, en general, cualquier

entregable generado durante el ciclo de vida del proyecto. Durante el desarrollo de las distintas actividades definidas en el proceso, son identificados y analizados todos aquellos elementos requeridos para soportar y mantener de manera efectiva el producto. Dichos requerimientos son documentados en el plan de mantenimiento, de cuyo cumplimiento depende la consecución del propósito arriba definido.

#### **5.6.1. Objetivos**

- Verificar que se tienen todos los artefactos requeridos para pasar el producto a operaciones
- Identificar el equipo que soportará y mantendrá el producto software
- Asegurar que el equipo seleccionado es el mejor calificado para desarrollar eficaz y eficientemente las labores de soporte del producto
- Establecer y dotar al equipo de soporte de los recursos necesarios para desempeñar las actividades de operación y mantenimiento del producto

#### **5.6.2. Actividades a Desarrollar**

- Identificar la Organización de Soporte y Mantenimiento
- Establecimiento y Revisión de Entregables y Actividades de Transición
- Planear el Despliegue del Producto Desarrollado
- Preparar y Evaluar la Organización de Soporte y Mantenimiento
- Revisar los Resultados de las Actividades de Capacitación

### 5.6.2.1. Identificar la Organización de Soporte y Mantenimiento

Esta actividad tiene como objetivo la identificación de la organización de soporte y mantenimiento que soportará la solución desarrollada en el proyecto y la definición del presupuesto requerido para apoyar las labores asociadas. En este sentido, la decisión más importante que debe tomarse es si el soporte de la solución desarrollada será asumido por la organización contratante o, si por el contrario, esta actividad será delegada a una organización externa. Los criterios que apoyan la toma de esta decisión son los mismos que se describen en la parte de tercerización del marco teórico de este trabajo (ver secciones 4.1.4 y 4.1.5) donde, en el caso de la transición para el soporte, toman más fuerza aquellos criterios que tienen que ver con costos y con las habilidades requeridas para desarrollar efectiva y eficientemente esta actividad. Adicionalmente, es importante anotar que la identificación de la organización de soporte y mantenimiento debe ser realizada en etapas tempranas del proyecto, preferiblemente antes que la solicitud de propuesta sea liberada. Esto con el fin que las labores de transición puedan ser llevadas a cabo en los tiempos establecidos, que el proveedor seleccionado ajuste su propuesta a esta decisión y que tenga conocimiento oportunamente de hacia quién deberá realizar la transición. La tabla 5.30 desarrolla una lista de chequeo que ayuda a acotar el personal requerido para soportar el producto:

<b>Roles a tener en cuenta para soportar el producto:</b>	
¿Quién supervisará los operadores del producto software (usuarios finales)?	<input checked="" type="checkbox"/>
¿Quién administrará el producto para mantenerlo disponible?	<input checked="" type="checkbox"/>
¿Quién generará los reportes de gestión y/o del producto? ¿Quién revisará dichos reportes?	<input checked="" type="checkbox"/>
¿Quién realizará la instalación de nuevo hardware o dispositivos de visualización (pantallas)?	<input checked="" type="checkbox"/>
¿Quién realizará la integración del producto con dispositivos de campo adicionales? (p.e., cuando el sistema requiere de una impresora para operar o de un sensor para recibir datos).	<input checked="" type="checkbox"/>

Tabla 5.30. Lista de chequeo para determinar el personal requerido para soportar el producto. Basado en [SAA98].

¿Quién corregirá los errores que pueda presentar el producto?	<input checked="" type="checkbox"/>
¿Quién realizará las actualizaciones del producto? (p.e., instalación de nuevas liberaciones, ampliación de funcionalidades, modificación del código fuente y similares).	<input checked="" type="checkbox"/>
¿Quién diagnosticará problemas y reparará hardware?	<input checked="" type="checkbox"/>
¿Quién(es) realizará(n) las capacitaciones requeridas para desempeñar las labores arriba descritas?	<input checked="" type="checkbox"/>

Tabla 5.30. Lista de chequeo para determinar el personal requerido para soportar el producto. Basado en [SAA98].

### 5.6.2.2. Establecimiento y Revisión de Entregables y Actividades de Transición

Con el fin de asegurar que todos los elementos requeridos para efectuar la transición se encuentran a punto, se deben llevar a cabo las siguientes tareas:

Identificar los Entregables a ser Transferidos

Es importante realizar una lista de los entregables que serán transferidos a la organización de soporte y mantenimiento. Elementos tales como código, documentación relativa al soporte y mantenimiento de la solución, sistemas de gestión de la configuración, material de capacitación, manuales de usuario y similares deben ser tenidos en cuenta.

Revisar los Entregables y Preparativos antes de la Transición

Antes que la transición sea efectuada, es importante cotejar el estado de los entregables y de las actividades relativas a la transferencia y, en caso de ser necesario, corregir las desviaciones encontradas respecto a las especificaciones establecidas y a lo planeado. Para dicho propósito es recomendable elaborar una lista de chequeo donde se incluyan los elementos y las actividades que deben ser consideradas de acuerdo a las circunstancias

particulares de cada proyecto. A continuación se presenta una lista que supone los aspectos más importantes a tener en cuenta:

- Usabilidad del software
- Completitud de la solución desarrollada
- Estado de la capacitación a los usuarios
- Completitud de los manuales de usuario
- Completitud de los planes y de la documentación de soporte y mantenimiento
- Estado de los preparativos y actividades de instalación del producto
- Preparación del software para la transición
- Estado de las actividades de preparación para la transición

Adicionalmente, es importante realizar una revisión de los aspectos concernientes a derechos de autor y licenciamiento que han sido previamente acordados de manera contractual con el proveedor, tales como copias de respaldo y número de instalaciones permitidas del producto desarrollado. Debido a que este tema sugiere serias repercusiones de tipo legal, la tabla 5.31 desarrolla una lista de chequeo que ayuda a clarificar las dudas resultantes respecto a la propiedad intelectual:

<b>Derechos Generales</b>	
¿Quién es el dueño del producto? ¿Qué derechos se supone que tiene sobre el?	<input checked="" type="checkbox"/>
¿Quién posee los derechos de autor del producto? ¿Qué derechos son?	<input checked="" type="checkbox"/>
¿Deben ser incluidos avisos de derechos de autor dentro del código fuente? ¿Quién debe ser mencionado como el propietario de los derechos de autor?	<input checked="" type="checkbox"/>

Tabla 5.31. Lista de chequeo para los aspectos relativos a la propiedad intelectual. Basado en [SAA98].

Derechos del Cliente	
¿Puede el cliente hacer copias adicionales del producto para uso interno dentro del proyecto? ¿Puede hacer uso de el en otros proyectos?	☑
¿Puede el cliente distribuir copias del producto a otras sedes de la organización que se encuentren dentro de la región donde se encuentra ubicado?	☑
¿Puede el cliente distribuir copias o licencias del producto a otras sedes de la organización que se encuentren por fuera de la región donde se encuentra ubicado?	☑
¿Habilitan estas licencias a las otras sedes para introducir cambios en el producto o mejorarlo, o solo es para propósitos de uso única y exclusivamente?	☑
¿Puede el cliente distribuir gratis copias del producto? ¿Puede obtener ganancias de ello?	☑
¿Puede el cliente introducir cambios en el software u obtener de el productos derivados?	☑
¿Puede el cliente revelar el código fuente del producto a otros proveedores, o puede permitirles introducir cambios sobre el mismo?	☑
Para los pasos anteriores, ¿qué partes del producto software pueden ser copiadas o distribuidas?, ¿el código fuente?, ¿el código objeto?, ¿la documentación?, ¿cuántas copias pueden realizarse?	☑
¿Tiene el cliente derecho a nuevas versiones del producto ("upgrades") desarrolladas por el proveedor?	☑
¿Existen partes del producto que requieran necesariamente hacer uso de otros productos software que no están cubiertos por los acuerdos de licenciamiento? Dichos productos pueden referirse a sistemas operacionales, motores de bases de datos, software base y similares. Para estos productos, al igual que para el software adquirido (o en proceso de adquisición), también pueden existir especificaciones acerca de el número de copias que pueden realizarse, utilizarse y distribuirse. ¿Se tiene lo anterior claro?	☑
¿Tiene el cliente acceso al código fuente del producto? Si lo tiene, ¿este acceso es a los archivos de compilación o simplemente a una lista de archivos?	☑
¿Tendrá acceso el cliente a las herramientas de soporte y al entorno de desarrollo que fue utilizado por el proveedor para compilar el producto? ¿Puede mantenerlo bajo control de la configuración?, ¿Puede probar sobre el?	☑
¿Tiene el cliente derechos sobre el material de capacitación?	☑
¿Tiene el cliente derechos sobre el ambiente de producción requerido para correr el producto o debe adquirirlo por separado de otros proveedores?	☑
¿Tiene el cliente acceso a la documentación relativa a la estructura de la base de datos y a los protocolos de interfases?	☑
¿Cuántos computadores pueden contener copias del producto? ¿Cuántos pueden utilizarlo? (Nota. Estos números pueden ser diferentes a los que se permiten para realizar copias de respaldo).	☑
¿Cuántos computadores pueden simultáneamente ejecutar o acceder al producto software? (Nota. En una red, todos los computadores pueden ser capaces de ejecutar una pieza de software o de acceder a una base de datos, pero los acuerdos de licenciamiento pueden restringir el número de estos que pueden hacerlo).	☑
¿Cuántos usuarios tienen licencia para ejecutar el producto software? ¿Cuántos lo pueden hacer simultáneamente?	☑

Tabla 5.31. Lista de chequeo para los aspectos relativos a la propiedad intelectual. Basado en [SAA98].

¿Puede el producto software ser ejecutado a través de una red? (Nota. En este punto existen dos opciones: el producto puede ser ejecutado de manera remota, es decir, sobre otra máquina donde la copia reside; o, una copia temporal podría realizarse sobre la máquina origen y correr el producto localmente.).	<input checked="" type="checkbox"/>
<b>Derechos del Proveedor</b>	
¿Puede el proveedor distribuir el producto software a otros clientes? Si es así, ¿Puede este cobrar por ello?	<input checked="" type="checkbox"/>
¿Puede el proveedor reutilizar piezas del producto software en otros proyectos?	<input checked="" type="checkbox"/>
¿Puede el proveedor adquirir los derechos de autor o patentar el producto software? ¿Puede este hacer lo propio para partes o piezas del mismo? Si es así, ¿tendrá el cliente que pagar por los derechos de autor?	<input checked="" type="checkbox"/>
¿Tiene el proveedor algún derecho sobre cualquier mejora o nuevas versiones desarrolladas sobre el producto por el cliente?	<input checked="" type="checkbox"/>

Tabla 5.31. Lista de chequeo para los aspectos relativos a la propiedad intelectual. Basado en [SAA98].

### 5.6.2.3. Planear el Despliegue del Producto Desarrollado

Con el propósito de minimizar contratiempos en el despliegue del producto desarrollado, es recomendable elaborar un plan que aglutine los pasos requeridos para llevar a cabo dicho despliegue y los prerrequisitos que deben ser tomados en consideración. Es importante tener en cuenta que las actividades relativas a la liberación del producto requieren un alto nivel de preparación y colaboración de la organización contratante y que la exitosa finalización de un proyecto software muchas veces se ve impactada por aspectos que se encuentran fuera del alcance del esfuerzo de desarrollo efectuado por el proveedor. El plan de despliegue debe describir como mínimo los siguientes aspectos:

#### 5.6.2.3.1. Responsabilidades

Se deben identificar las responsabilidades tanto del proveedor como de la organización contratante en las actividades de despliegue. Es de particular

importancia la descripción del grado en el que la organización contratante estará involucrada en las pruebas de aceptación y los procedimientos para manejar discrepancias después de la instalación del producto.

#### 5.6.2.3.2. Cronograma

Debe elaborarse un cronograma en el cual se describan las actividades, los tiempos y los hitos que se ven involucrados en el despliegue de la solución desarrollada. Cabe anotar que los hitos que se propongan deben estar en sintonía con los hitos globales del proyecto. Algunas de las actividades que pueden ser tomadas en cuenta dentro de este cronograma son:

- Desarrollo del plan de despliegue
- Puesta a punto de la documentación de soporte y mantenimiento
- Puesta a punto de los manuales de usuario
- Ejecución y documentación de las pruebas de aceptación de la instalación del producto
- Desarrollo del paquete de instalación (proveedor)
- Elaboración de los medios de instalación proveedor
- Capacitación a usuarios finales
- Capacitación a la organización de soporte y mantenimiento

#### 5.6.2.3.3. Descripción de Recursos y Facilidades

Es importante que se documenten todas aquellas facilidades que se requieran para llevar a cabo la instalación del producto. Estas facilidades pueden incluir requerimientos de energía, cuartos con características especiales (por ejemplo, un salón con piso falso), herramientas para mantener la privacidad o seguridad de la instalación y similares.

#### 5.6.2.3.4. Descripción de Prerrequisitos

Deben describirse con exactitud todos aquellos requisitos de hardware y software base que deben satisfacerse para que la instalación y el correcto funcionamiento del producto desarrollado sean exitosos. Esta descripción debe incluir los modelos, versiones y configuraciones a aplicar. Adicionalmente, debe proveerse información acerca de las casas que manufacturan estos productos y de las especificaciones de licenciamiento.

#### 5.6.2.3.5. Documentación de Soporte y Mantenimiento

Debe elaborarse una descripción completa de la documentación de soporte que será liberada con el producto desarrollado. Esta documentación puede hacer referencia a descripciones de diseño, casos y procedimientos de prueba, manuales de usuario, especificaciones para el mantenimiento del producto, procedimientos de soporte y similares.

#### 5.6.2.3.6. Organización de Soporte y Mantenimiento

Si las necesidades particulares del proyecto así lo requieren, debe realizarse una descripción del personal requerido para soportar el producto, los roles que cumplirían y las habilidades requeridas para desempeñar las actividades de soporte y mantenimiento que se definan.

#### 5.6.2.3.7. Capacitación

Es importante que se realice una descripción del plan de capacitación que se empleará para adelantar las labores de entrenamiento tanto de usuarios finales como del personal que soportará y mantendrá el producto. En el caso de los usuarios finales, es necesario que se incluya la estrategia que se empleará

para que estos puedan adaptar el producto de acuerdo a las necesidades particulares que tengan.

#### **5.6.2.4. Preparar y Evaluar la Organización de Soporte y Mantenimiento**

Esta actividad tiene como objetivo la personalización y transición de los procesos necesarios del equipo de desarrollo al equipo de soporte y mantenimiento. Para dicho propósito las siguientes tareas deben ser llevadas a cabo:

##### **5.6.2.4.1. Elaborar un Plan de Mantenimiento**

Con el fin que el producto software desarrollado se mantenga siempre disponible y que su funcionamiento cumpla de manera permanente con los requerimientos especificados, es necesaria la elaboración de un plan de mantenimiento. La tabla 5.32 describe los elementos que como mínimo deben ser documentados en el plan:

<b>Como mínimo el plan de mantenimiento debe contener:</b>	
Identificar los miembros del equipo de soporte y mantenimiento, sus respectivas responsabilidades y la manera en que están organizados.	<input checked="" type="checkbox"/>
Describir las interacciones que el equipo de soporte y mantenimiento tendrá con otros grupos dentro y fuera de la organización contratante.	<input checked="" type="checkbox"/>
Describir los recursos destinados a las labores de mantenimiento. Estos recursos pueden tomar la forma de equipos, herramientas, material, documentación, facilidades y similares.	<input checked="" type="checkbox"/>
Especificar los fondos y las asignaciones presupuestales destinadas a las labores de soporte y mantenimiento.	<input checked="" type="checkbox"/>

Tabla 5.32. Lista de chequeo del plan de mantenimiento. Basado en [SVE01].

Determinar el alcance de las labores de mantenimiento y la forma en que estas serán coordinadas.	<input checked="" type="checkbox"/>
Describir los procedimientos establecidos para el reporte de problemas y de las labores de mantenimiento que han sido efectuadas.	<input checked="" type="checkbox"/>
Establecer cómo el análisis de problemas y modificaciones sobre el producto será desarrollado.	<input checked="" type="checkbox"/>
Describir los procedimientos establecidos para la implementación de modificaciones sobre el producto desarrollado.	<input checked="" type="checkbox"/>
Describir cómo las actividades de mantenimiento serán revisadas y aceptadas	<input checked="" type="checkbox"/>
Establecer procedimientos para la desinstalación o migración del producto.	<input checked="" type="checkbox"/>
Establecer los procedimientos y las actividades para hacer gestión de la configuración sobre el producto y los entregables relacionados.	<input checked="" type="checkbox"/>
Establecer los procedimientos y las actividades para la continuación de las labores de la administración de requisitos.	<input checked="" type="checkbox"/>
Especificar los estándares, métodos, prácticas y convenciones que serán seguidas.	<input checked="" type="checkbox"/>

Tabla 5.32. Lista de chequeo del plan de mantenimiento. Basado en [SVE01].

#### 5.6.2.4.2. Evaluar las Habilidades y los Recursos del Equipo de Soporte y Mantenimiento

En aras de garantizar la estabilidad futura del producto desarrollado, es recomendable evaluar las habilidades y los recursos con los que cuenta el equipo que deberá soportar y mantener dicho producto. A continuación se describen algunos de los parámetros que deben ser evaluados:

<b>Recursos y habilidades del equipo de soporte y mantenimiento:</b>	
¿Tiene el equipo los recursos hardware y software apropiados?	<input checked="" type="checkbox"/>
¿Cuenta el equipo con los fondos y/o el presupuesto necesario para mantener y soportar el producto?	<input checked="" type="checkbox"/>
¿Cuenta el equipo con el suficiente personal para soportar y mantener el producto?	<input checked="" type="checkbox"/>
¿Cuenta el equipo con un plan de mantenimiento?	<input checked="" type="checkbox"/>
¿Dispone el equipo de la documentación técnica, los procesos y los procedimientos necesarios?, ¿Los procesos y procedimientos son efectivos y eficientes?	<input checked="" type="checkbox"/>
¿Cuenta el equipo con un sistema de gestión de la configuración capaz de soportar el producto y sus entregables?	<input checked="" type="checkbox"/>
¿El equipo se encuentra debidamente capacitado?	<input checked="" type="checkbox"/>
¿Los miembros del equipo son competentes para adelantar las labores de soporte y mantenimiento?	<input checked="" type="checkbox"/>

Tabla 5.33. Lista de chequeo de los recursos y las habilidades del equipo de soporte y mantenimiento. Basado en [SAA98].

#### **5.6.2.5. Revisar los Resultados de las Actividades de Capacitación**

Esta actividad tiene como propósito la verificación que los acuerdos realizados en cuanto a capacitación de usuarios han sido cumplidos. Algunas de las tareas que deben haberse satisfecho en esta etapa son:

##### **5.6.2.5.1. Determinar la Audiencia que Asistirá a las Capacitaciones**

En esta tarea se busca establecer los usuarios finales que deberán asistir a las sesiones de capacitación o a cualquier otra técnica de entrenamiento que se haya seleccionado. Es importante que se entienda el perfil de los asistentes, es decir, su nivel académico y profesional, su experiencia, su edad, el tiempo que llevan en la organización y demás elementos que pueden ayudar a hacer una personalización más efectiva de la capacitación. De igual modo, el perfil y la cultura de la organización contratante brindan algunos elementos de juicio que pueden ayudar a perfilar de mejor manera la estrategia de capacitación. Adicionalmente, deben establecerse todos aquellos prerrequisitos que los asistentes deben satisfacer para que las capacitaciones sean realmente efectivas.

##### **5.6.2.5.2. Definir los Objetivos de la Capacitación**

Esta tarea tiene como propósito establecer las características específicas del producto software que los asistentes necesitan aprender o de las cuales pueden sacar provecho. En muchos casos no es posible que en una sesión de entrenamiento se cubran todos los detalles acerca del funcionamiento y operación del producto y por eso se hace necesario establecer el grado de conocimiento que los asistentes a la capacitación deberían lograr. Debe tenerse en cuenta que, para algunas funcionalidades, una discusión a nivel

introductorio puede ser suficiente, mientras que otras pueden ser críticas para los usuarios y deben tratarse con un nivel de detalle más elevado.

#### 5.6.2.5.3. Determinar el Tipo de Capacitación a Aplicar

La determinación del tipo de capacitación a aplicar es una de las decisiones más importantes en el contexto de esta actividad. Existen muchas formas de adelantar labores de capacitación y entrenamiento y todas ellas dependen de la situación particular de cada proyecto y de las necesidades puntuales de los usuarios finales. De igual modo y como se mencionó anteriormente, en este punto también entran a jugar un papel determinante el perfil de los asistentes y la cultura organizacional del ente contratante. Algunas de las opciones que pueden considerarse son:

- Tutoriales en línea
- Cursos formales de capacitación
- Talleres de capacitación
- Técnicas colaterales: notas en el producto, “tips”, “wizards” y similares
- Entrenamiento por Internet
- Ayuda y soporte telefónico

#### 5.6.2.5.4. Determinar la Duración de la Capacitación

En caso que la técnica de capacitación seleccionada sean los cursos formales de capacitación, debe decidirse el número de sesiones a realizarse, cuanto durarán las mismas y el número máximo de asistentes permitidos. De igual modo y, en aras de cubrir mejor las necesidades de los usuarios, debe considerarse el desarrollo y ofrecimiento de una familia de cursos en vez de uno solo.

#### 5.6.2.5.5. Desarrollar los Materiales de Capacitación

El desarrollo de los materiales de capacitación debe concentrarse en aquellos materiales que serán utilizados durante las sesiones formales de capacitación. Si se están utilizando los talleres de capacitación como estrategia, estos materiales pueden tomar la forma de diapositivas o de implementos que se requieran para realizar los ejercicios que se propongan. Si los materiales de capacitación no serán utilizados solamente por el instructor encargado de dictar las sesiones de capacitación, debe considerarse el desarrollo de guías para el instructor y de especificaciones para el desarrollo de estos materiales. De igual modo, debe considerarse la elaboración de una lista de referencias a libros, artículos, sitios de Internet y similares, que puedan ayudar a los asistentes a clarificar o profundizar en temas o conceptos que sean manejados directa o indirectamente por ciertas funcionalidades del producto software. Finalmente y, con el propósito que los conocimientos transmitidos perduren, es recomendable entregar a los asistentes copias impresas de los materiales de capacitación.

## 6. CONSIDERACIONES FINALES

Después de haber tenido la oportunidad de haber desarrollado un proyecto de adquisición, es importante ahondar en la experiencia y en las enseñanzas adquiridas. ¿Qué puede ser realizado de mejor manera en proyectos de adquisición futuros? La solución a la anterior pregunta, busca fundamentalmente que en un proyecto futuro se evite al máximo cometer los mismos errores, y mejorar los procesos de adquisición actuales mediante la obtención oportuna de retroalimentación. Lo anterior también incluye la documentación de los distintos datos arrojados por el proyecto y del desempeño obtenido por el proveedor para referencias futuras.

En concreto, en las sesiones de finalización y cierre del proyecto de adquisición la organización cliente debería verse interesada en alguno de los siguientes propósitos [KEN06]:

- Recabar y analizar los distintos datos arrojados por el proyecto con el fin de entender los procesos existentes, mejorar los procesos de adquisición, obtener información base para referencias futuras, o detectar necesidades de capacitación.
- Analizar las causas de los problemas e incidentes ocurridos, recoger el consenso global acerca de las mismas, documentarlas, y crear una base de conocimiento para el apoyo y referencias futuras.
- Para reparar perjuicios causados a los integrantes del equipo del proyecto.
- ¡Para disfrutar de la realización del proyecto y de los logros obtenidos!

Es importante tener en cuenta que con el propósito de evitar que las actividades de finalización y cierre del proyecto no se vean sesgadas por un mal ambiente, debe crearse una atmósfera agradable donde los participantes sientan que se trata de una experiencia de aprendizaje y no de una sesión de asignación de culpas. Después de establecida esta atmósfera, deben obtenerse los datos que serán analizados durante las sesiones de finalización los cuales, por lo general, hacen referencia a los esfuerzos realizados, los recursos consumidos, la calidad de los productos obtenidos y similares. En este sentido, cada participante debe recurrir a sus propias experiencias para realizar las aportaciones que vengan al caso.

Cuando todo se encuentre listo, el análisis de la información recopilada y de los incidentes ocurridos debe realizarse. Este análisis debe concentrarse en los siguientes aspectos:

- Identificación de buenas y malas prácticas de adquisición
- Las cosas que se hicieron bien y las cosas que se hicieron mal
- El consumo del presupuesto asignado al proyecto y/o los sobrecostos generados. ¿Por qué se generaron?
- Ceñimiento a los planes elaborados y cumplimiento del cronograma. ¿Se tuvieron retrasos?, ¿por qué se generaron?, ¿hubo desviaciones respecto a lo especificado en los planes?, ¿por qué?
- Incidentes ocurridos y las razones de los mismos
- ¿Qué fue logrado?, ¿cuál fue la calidad del producto liberado?

- Establecer las dificultades específicas que se tuvieron y como fueron manejadas
- Establecer el desempeño del proveedor y la conformidad con los requerimientos
- ¿Los procesos de adquisición cumplieron sus objetivos?, ¿fueron óptimos?, ¿necesitan ser ajustados?, ¿Qué recomendaciones existen en cuanto a la personalización de los mismos?
- Revisar las sugerencias de mejoramiento
- ¿Qué debe ser realizado de distinta manera la próxima vez?

Debe tenerse en cuenta que la retroalimentación del proveedor es fundamental, y que si éste no fue incluido en las sesiones de finalización, es importante que se programe al menos una en la cual participe. Por último, y como cierre definitivo, toda la información recogida en estas sesiones, las enseñanzas aprendidas, las recomendaciones dadas y los datos acerca del desempeño del proveedor, deben ser documentadas y almacenadas en un lugar donde de manera segura pueda ser accedida en los proyectos venideros como base para el apoyo de los mismos.

## 7. CONCLUSIONES

- Los proyectos donde se delega la responsabilidad de desarrollar el producto software en un proveedor, o proyectos de adquisición son, por naturaleza, distintos a otro tipo de proyectos. El software es intangible, no se fabrica, no se deteriora con el tiempo, las metodologías de desarrollo y los procesos existentes aún son inmaduros, y los costos de mantenimiento son más elevados que los del hardware.
- Situaciones indeseables como errores de planificación, sobrecostos, retrasos y productos software que no satisfacen las necesidades de clientes y usuarios, son muy comunes en los proyectos de adquisición. Por tal motivo, es preciso aplicar tanto técnicas ingenieriles como técnicas de dirección a este tipo de proyectos.
- Es utópico pensar en una secuencia de acciones que dirijan paso a paso todos los proyectos delegados de desarrollo de software; al contrario, muchas de las actividades de adquisición deben llevarse a cabo de manera simultánea, y su utilización, personalización, y la forma en que se desarrollen, depende del entorno y de las características particulares de cada proyecto y varían de organización a organización.
- El tema del software cobra cada vez mas importancia para las organizaciones modernas, ya que al apoyar de manera directa los procesos de negocio, los procesos productivos y/o los procesos administrativos, se pueden obtener grandes ventajas competitivas o comparativas. Es por eso, que las consideraciones concernientes a la calidad del software y en especial, al aseguramiento de la calidad, deben desempeñar un serio papel en los proyectos de adquisición.

- Muchas organizaciones yerran en su intento por producir software de calidad. Muchos de estos errores se originan en la absoluta focalización de las organizaciones en los aspectos relativos a la calidad del producto. Es por eso que debe tenerse en cuenta y aplicarse, las prácticas promovidas por los modelos de calidad mas modernos y reconocidos, que indican que una parte fundamental de un buen sistema de aseguramiento de la calidad, es la calidad de los procesos ejecutados para desarrollar el software.
- Existen en la literatura un sin número de metodologías, estándares, marcos de proceso y modelos que guían la forma en que deben llevarse los procesos de desarrollo de software, todos con el objetivo y la intención de producir software mayor calidad. La escogencia de un estándar en particular depende de la intención de uso; cada estándar define una orientación específica y lo ideal es escoger el que mejor se acomode a los objetivos organizacionales.
- La estrategia desarrollada en este proyecto, es decir, el marco de procesos, se encuentra fuertemente enfocada a organizaciones cuyo objeto de negocio no es el de desarrollar software, pero que requieren de el para soportar sus diferentes procesos. Este modelo, puede brindar a ese tipo de organizaciones unas bases sólidas para desempeñar de manera eficaz y eficiente los proyectos de adquisición emprendidos y, no menos importante, asegurar que los productos software obtenidos sean de óptima calidad.
- Todas las actividades definidas en los procesos de adquisición, sin exclusión, se convierten en factores críticos para el éxito del proyecto. En particular, la conformación de un grupo de profesionales que posean las habilidades requeridas para afrontar los retos propuestos en un proyecto delegado de desarrollo de software, es esencial; en caso que el

equipo conformado no posea las habilidades requeridas, estas deben ser buscadas en otros miembros de la organización o deben ser subcontratadas.

- El plan de adquisición es uno de los artefactos mas importantes con los que cuenta el proyecto, ya que de el se desprende la estrategia con la que será atacado el mismo. Debe tenerse en cuenta, que este plan es un documento vivo, y que puede ser refinado añadiendo más información o modificando la que ya contiene.
- Una organización cliente debe hacer estimados de los tiempos y de los costos involucrados en el proyecto. Sin embargo, debe tenerse en cuenta que dichas estimaciones no deben ser realizadas única y exclusivamente por el cliente; al contrario, y de ser posible, entre más estimaciones se tengan y en especial, si se involucra al proveedor en esta actividad y se resuelven diferencias con el, se podrá obtener una planificación mas acertada del proyecto que finalmente redundará en la satisfacción de las partes y, no menos importante, en la mitigación de sobrecostos y retrasos.
- Es muy importante tener en cuenta que, sea cual sea el estándar, la metodología o el marco de procesos a seguir, todo proyecto requiere adaptar la guía escogida; el proyecto que se ciñe estrictamente a cualquiera de estas opciones, sin análisis, sin sensatez y sin configuración, muy probablemente consiga fracaso. Por tal motivo, es imprescindible personalizar, adaptar o configurar el estándar seleccionado, de tal modo que se pueda ajustar adecuadamente a las características particulares del proyecto de adquisición.
- Respecto a los modelos de calidad estudiados, es importante que para un proyecto software se escoja un estándar que trate los aspectos

específicos que conciernen al tema del software. En este sentido, se puede decir que modelos como el propuesto por la ISO 9000 son demasiado genéricos, simples, generales y no ofrecen una estrategia que guíe paso a paso en su implementación. Por otro lado, modelos como los propuestos por CMMI y la ISO 15504, son fuertemente enfocados al caso específico del software y los sistemas de información. La ISO 15504 es uno de los estándares mas consensuados y probados que existen en modelos de su tipo, pero tiene en su contra que las valoraciones son difíciles de realizar y que no posee una estructura propia de procesos. CMMI es el estándar que en la actualidad goza de mayor prestigio y brinda a sus usuarios una estructura propia de procesos; pese a lo anterior, debe considerarse que este modelo es difícil de entender, tiende a ser prescriptivo y requiere de una fuerte inversión económica.

- Los requisitos son uno de los factores mas críticos en cualquier proyecto de desarrollo de software, y de su buena especificación depende en gran medida el éxito del mismo. Esta actividad es clave, ya que además de contener y crear un filtro para las necesidades organizacionales, los requisitos pueden ser empleados para estimar el tamaño, los plazos y los costos del proyecto; adicionalmente, estos se convierten en el insumo principal para establecer los criterios bajo los cuales la organización cliente recibirá el producto, y tienen serias implicaciones en la forma en que el proveedor diseña y desarrolla este mismo.
- Los criterios de costos a la hora de seleccionar un proveedor, por lo general siempre se ven favorecidos por organizaciones cuyos procesos de selección no han sido realmente madurados; debe tenerse en cuenta, que otros criterios como las habilidades técnicas, el enfoque metodológico, y el desempeño pasado, pueden llegar a ser

determinantes para el éxito del proyecto, y que asumir los costos como criterio único o mas relevante puede llegar a ser mucho mas costoso.

- Muchos proyectos de adquisición fracasan porque la forma en que se realizan los pagos al proveedor y el monto de los mismos no satisfacen a alguna de las partes. En este sentido, es de vital importancia que se defina el tipo de contrato que, de manera equilibrada, pueda adecuarse al beneficio de ambas partes. Para este propósito, y como elemento esencial, debe recurrirse al alcance definido para el proyecto; debe tenerse en cuenta, que para proyectos donde el alcance es claro y se encuentra bien delimitado, los contratos de costos fijos son recomendables; y para proyectos donde el alcance es volátil o difícil de definir y donde se requiere de mayor flexibilidad, los contratos de costos reembolsables son mas adecuados.
- Dada sus connotaciones legales, el tema de la propiedad intelectual y del licenciamiento, es de suma relevancia en el contexto de los proyectos de adquisición. Lamentablemente, este es uno de los temas mas confusos y mas difíciles de manejar para las organizaciones actuales. Es por eso, que es muy recomendable recurrir a la asistencia profesional de abogados, de tal modo que estos asuntos pueden ser aclarados y que posteriormente no se vaya incurrir en algún tipo de delito. Los asuntos que conciernen a la propiedad intelectual, derechos de autor, licenciamiento y similares, deben ser explícitamente especificados de manera contractual, y las partes deben estar absolutamente claras de lo que realmente significan y deben de estar de acuerdo con ellos.
- Los criterios de aceptación son, tal vez, el elemento mas crítico e importante en un proyecto de adquisición. Estos criterios protegen contractualmente al cliente de situaciones indeseables como productos

software que no cumplen los requerimientos establecidos, artefactos incompletos o no entregados, e incumplimientos de las entregas en las fechas acordadas. Es por eso, que se debe hacer un gran esfuerzo en especificar de la manera mas clara y mas detallada posible los criterios de aceptación y, además, debe asegurarse que tanto cliente como proveedor los tienen absolutamente claros y que están de acuerdo con ellos. Adicionalmente, debe tenerse en cuenta que los criterios de aceptación por si solos son inefectivos; es imprescindible que también se especifiquen los procedimientos mediante los cuales se realizará la aceptación de los distintos entregables; planes de pruebas, casos de prueba, procedimientos de prueba, criterios de aceptación y similares, conforman una estrategia ideal, hacen parte integral de la Evaluación del Producto y van en pro tanto de los intereses de la organización cliente, como de la protección de derechos del proveedor.

- A pesar que un producto software pueda cumplir con todas las especificaciones requeridas, muchas organizaciones fallan en realizar una buena transición de los artefactos que el producto asocia. A pesar de la importancia inherente al producto, dado que es el resultado mas valuable del proyecto, alrededor del mismo se tejen una serie de actividades y servicios con son vitales para su supervivencia. Muchos de estos servicios se construyen en las labores de transición del producto al ambiente operativo donde deberá ser soportado, donde se hace imprescindible conformar y capacitar un equipo que cuente con las habilidades requeridas para desempeñar este tipo de labores. Conjuntamente, y no menos importante, es imperativo que las actividades de soporte sean planeadas de manera temprana en el proyecto casi en paralelo con las actividades de desarrollo, y que se asignen los recursos requeridos, tanto financieros como materiales, para que el equipo pueda desempeñar estas labores sin mayores inconvenientes. Una anotación final que debe tenerse muy en cuenta

respecto al último punto: durante el ciclo de vida del producto software, las actividades de soporte, por lo general, consumen mucho más presupuesto que las mismas actividades de desarrollo.

- Es muy importante, y mas bajo el contexto de la tercerización de las actividades de desarrollo de software, que la organización contratante haga una adecuada gestión del proyecto de adquisición. Al delegar estas actividades en un proveedor, al cliente no le queda mas que concentrar gran parte de sus esfuerzos en adelantar aquellas tareas tendientes a asegurar que los compromisos adquiridos están siendo cumplidos. Esta labor, después de haber realizado la planeación del proyecto, la especificación de requisitos, y la selección y contratación del proveedor, es la que mas relevancia toma a nivel del cliente y la que más se prolonga en el tiempo. Por tal motivo, es imprescindible que se determinen con exactitud los mecanismos a través de los cuales el progreso del proyecto será medido, como se manejarán las comunicaciones, como se adelantarán los pagos, como se reportarán y escalarán los problemas, como se harán las modificaciones sobre el alcance, tiempo y costos definidos, y, aún mas importante, las condiciones bajo las cuales se dará feliz término al contrato. Para tener éxito en esta dura labor, el proveedor debe estar totalmente involucrado en el desarrollo de las especificaciones arriba descritas y se debe poner a disposición los recursos requeridos para adelantar las labores de seguimiento.
- Después de la finalización del proyecto de adquisición, y en aras de no repetir los mismos errores en proyectos futuros, es muy importante obtener retroalimentación acerca de los distintos sucesos ocurridos dentro del proyecto. Esta retroalimentación debe hacer referencia a puntos clave como los incidentes ocurridos, como fueron manejados, la calidad de los productos obtenidos, el desempeño del proveedor, el

ceñimiento a los planes, la utilización de los recursos asignados, las buenas y malas prácticas, y la conformidad con los requerimientos. Esta información se convierte en una base de conocimiento sobre la cual se pueden entender los procesos seguidos y, a partir de ella, se pueden mejorar los procesos de adquisición: mejoramiento continuo.

- Los procesos propuestos en el marco de procesos otorgan una amplia cobertura a las organizaciones cliente, ya que parten de una especificación preliminar del proyecto y recorren todo el ciclo de vida del mismo hasta asegurar que una debida transición del producto desarrollado ha sido realizada.
- Finalmente, de manera implícita la propuesta desarrollada define un sistema de aseguramiento de la calidad, debido a que el seguimiento juicioso pero a la vez sensato de los objetivos, actividades, tareas, entregables y listas de chequeo definidas en cada uno de los procesos de adquisición, establecen los límites y las pautas bajo las cuales un producto software puede ser producido con calidad.

## 8. BIBLIOGRAFIA

- [GOR95]** GOLDBERG, A., y RUBIN, K., Succeeding with Objects: Decision Frameworks for Project Management. Addison-Wesley, 1995. 542 p.
- [SEI02]** SOFTWARE ENGINEERING INSTITUTE. CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS, V1.1) Staged Representation. CMU/SEI-2002-TR-012 ESC-TR-2002-012, 2002.
- [WIK06]** WIKIPEDIA. Outsourcing [en línea]. Disponible en web: <<http://en.wikipedia.org/wiki/Outsourcing>> [Consulta: 15 de Marzo de 2006]
- [CAC04]** CONFEDERACIÓN ALEMANA DE COOPERATIVAS. Outsourcing en las Cooperativas de Ahorro y Crédito de América Latina [Documento Electrónico], 2004.  
< <http://www.dgrv.org/docs/estudiosDGRVn2.pdf>>
- [HEY02]** HEYWOOD, J, Brian. El dilema del Outsourcing: La búsqueda de la competitividad. Ed. Financial Times - Prentice Hall, 2002. 202 p.
- [ROR96]** ROTHERY, Brian y ROBERTSON, Ian , "Outsourcing", Limusa, México, 1996.
- [BAZ06]** BAZIOTOPOULOS, A, Leonidas. Logistics Innovation and Transportation. Work-in-Progress Conference paper, EuroCHRIE Thessaloniki, 2006.

- [FGR98]** FLOREZ, Diana, GOMEZ, William y RESTREPO, Juan. Outsourcing Informático. Medellín, 1998. Trabajo de grado. Universidad EAFIT. Departamento de Organización y Gerencia.
- [LOP06]** LÓPEZ, Evelin. Objetivos Organizacionales [en línea]. Disponible en web: <<http://www.monografias.com/trabajos15/objetivos-organizacionales/objetivos-organizacionales.shtml>>. [Consulta: 15 de Marzo de 2006]
- [CAE05]** CARDONA, Raúl y ESTRADA, Jaime. Estudio de Factibilidad para la Implementación de un Sistema de Outsourcing en las Empresas del Sector Público. Medellín, 2005. Trabajo de grado. Universidad EAFIT. Facultad de Administración.
- [MOG06]** MONTES, Maria y GONZÁLES, Karen. Análisis del Outsourcing de TI en el Sector Financiero de la Ciudad de Medellín. , 2006. Trabajo de grado. Universidad EAFIT. Departamento de Informática y Sistemas.
- [MOL06]** EL MOLINO. La Utilidad del Outsourcing o Subcontratación en las ORGANIZACIONES [en línea]. Disponible en web: <<http://www.paisrural.org/molino/11/outsourcing.htm>>. [Consulta: 13 de Junio de 2006]
- [HIS05]** HISPASEC CORPORATE. Los riesgos en el outsourcing de procesos TI [en línea], 2005. Disponible en web: <<http://www.hispasec.com/corporate/noticias/56>>. [Consulta: 11 de Mayo de 2006]
- [EUR04]** EUROPEAN INFORMATION TECHNOLOGY EXCHANGE. Key outsourcing terms [en línea], 2004. Disponible en web: <[http://www.euroitx.com/content/a\\_20052004\\_01.php](http://www.euroitx.com/content/a_20052004_01.php)>. [Consulta: 11 de Mayo de 2006]

- [PER04]** PERFECT IMAGE. IT Outsourcing, white paper [Documento Electrónico], Noviembre 2004. <<http://www.perfect-image.co.uk/downloads/whitepapers/IT%20Outsourcing%20White%20Paper%2011%2004.pdf>>
- [LEM06]** LEMON, Sumner. La subcontratación requiere una planeación cuidadosa [en línea]. Disponible en web: <<http://www.cioal.com/cioaldocs.nsf/0/2B266DEA8BA83DD60025704500512CB4>>. [Consulta: 11 de Mayo de 2006]
- [GEO04]** GEOFF, Bessin. The Business Value of Software Quality: Technical Discussion of Software Quality, White Paper [Documento Electrónico], Junio 2004.
- [JMJ90]** J. M, Jurán. Jurán y el liderazgo para la calidad. Un manual para directivos. Edición Díaz de Santos, S.A., México, 1990. 363 p.
- [ISO00]** NC ISO 9000:2000. Normas para la gestión de calidad. Parte 1, Fundamentos y vocabulario.
- [CRO87]** CROSBY, Philip B.; La Calidad no cuesta: el arte de asegurar la calidad; Mexico.D.F: Continental, 1987.
- [GIL74]** GILMORE, Harold. Product conformance cost, Quality Progress. Vol.7, No.5, 1974.
- [FEI91]** FEIGENBAUM, Armand V.Total Quality Control McGraw-Hill Inc. USA, 1991.
- [ATT56]** AT&T. TechnologiesStatistical Quality Control Handbook, AT&T Technologies. Del Mar Printing, Charlotte, N.C., 1956.
- [LOR04]** LORA, Norman M. La Calidad Como Factor Diferenciador En la Fabricación de Software. En: JORNADAS DE INGENIERÍA DEL SOFTWARE Y BASES DE DATOS. (9º : 2004 : Malaga).

Ponencia de la IX Jornadas de Ingeniería del Software y Bases de Datos. Málaga: Caribus Technology, 2004. 13 p.

- [RES06]** RESPONSABILIDAD INTEGRAL. Cual es el Valor Agregado [en línea]. Disponible en web: <[http://responsabilidadintegral.org/nuevo/valor\\_agregado.php](http://responsabilidadintegral.org/nuevo/valor_agregado.php)>. [Consulta: 14 de Mayo de 2006]
- [PRE02]** Pressman, R.S., Ingeniería del Software. Un enfoque práctico. Mc Graw Hill, 2002.
- [SCA06]** SCALONE, Fernanda. Estudio Comparativo de los Modelos y Estándares de Calidad del Software. Buenos Aires, 2006. Trabajo de grado. Universidad Tecnológica Nacional. Facultad Regional Buenos Aires.
- [CUE99]** CUEVA, Juan M. Calidad del Software, white paper [Documento Electrónico], Octubre 1999. [http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad\\_software.pdf](http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad_software.pdf)
- [ATL06]** ATlassian JIRA. Software Quality Assurance [en línea]. Disponible en web: <<http://jira.atlassian.com/secure/attachment/17146/sqa+activities.txt>>. [Consulta: 28 de Junio de 2006]
- [CAM78]** CAVANO, Joseph y MCCALL, James. A FRAMEWORK FOR THE MEASUREMENT OF SOFTWARE QUALITY, white paper [Documento Electrónico], Noviembre 1978.
- [CER06]** CERVERA, Angel. El Modelo de McCall como Aplicación de la Calidad a la Revisión del Software de Gestion Empresarial [en línea]. Disponible en web: <<http://www.monografias.com/trabajos5/call/call.shtml>>. [Consulta: 16 de Marzo de 2006]

- [UNO03]** Unosson, Johannes. Where To Find Successful Software Quality Management: A search for capable software engineering in theory and practice. Wien, 2003. Trabajo de grado. Universität Wien Fakultät für Technische Naturwissenschaften und Informatik.
- [SAN03]** SANDOVAL, Mauricio. Calidad y Desarrollo Organizacional a través de la Certificación ISO 9000, white paper [Documento Electrónico], Agosto 2003. <<http://www.ripit.granma.inf.cu/biblioteca%20Digital/calidad/calidad%20y%20desarrollo.pdf>>
- [VRR06]** VILLA, Manuel, RUIZ, Mercedes y RAMOS, Isabel. Modelos de Evaluación y Mejora de Procesos: Análisis Comparativo, white paper [Documento Electrónico]. <<http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-120/paper4.pdf>>. [Consulta: 22 de Mayo de 2006]
- [SUI06]** SELECTION AND USE OF THE ISO 9000:2000 FAMILY OF STANDARDS [en línea]. Disponible en web: <[http://www.iso.org/iso/en/iso9000-14000/understand/selection\\_use/selection\\_use.html](http://www.iso.org/iso/en/iso9000-14000/understand/selection_use/selection_use.html)>. [Consulta: 13 de Junio de 2006]
- [ITS01]** ISO/TC 176/SC 2/N 544R. Orientación acerca del enfoque basado en procesos para los sistemas de gestión de la calidad. Madrid: AENOR. Format PDF, 2001.
- [JMJ90]** J. M, JURÁN. Jurán y el liderazgo para la calidad. Un manual para directivos. Edición Díaz de Santos, S.A., México, 1990. 363 p.
- [BUP06]** BUSCA PORTAL. ISO 9001 - Norma de Calidad, Gestión de la Calidad o Excelencia [en línea]. Disponible en web: <[http://www.buscarportal.com/articulos/iso\\_9001\\_gestion\\_calidad.html](http://www.buscarportal.com/articulos/iso_9001_gestion_calidad.html)> [Consulta: 13 de Junio de 2006]

- [ISP06]** ISO 9000 2000 PRINCIPLES [en línea]. Disponible en web: <[http://www.buscarportal.com/articulos/iso\\_9001\\_gestion\\_calidad.html](http://www.buscarportal.com/articulos/iso_9001_gestion_calidad.html)> [Consulta: 13 de Junio de 2006]
- [ISS06]** ISO SPICE [en línea]. Disponible en web: <<http://www.isospice.com>> [Consulta: 8 de Abril de 2006]
- [WIS06]** WHY AN INTERNATIONAL STANDARD [en línea]. Disponible en web: <<http://www.sqi.gu.edu.au/spice/>> [Consulta: 8 de Abril de 2006]
- [ISO03]** The International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC FDIS 15504-2: Information Technology - Process Assessment – Part 2 - Performing an Assessment, Geneve: ISO, 2003.
- [ALG97]** AMESCUA, Antonio, LLORÉNS, Juan y GARCÍA, Ángel. SPICE: UN MARCO PARA LA EVALUACIÓN DE PROCESOS SOFTWARE, white paper [Documento Electrónico], Agosto 1997. <  
<  
<http://www.ie.inf.uc3m.es/grupo/Investigacion/LineasInvestigacion/Articulos/spice.doc>> [Consulta: 22 de Marzo de 2006]
- [ISU06]** ISO 9001:2000 UPDATE [en línea]. Disponible en web: <<http://www.qualitydigest.com/Oct00/html/iso9000.html>> [Consulta: 15 de Junio 2006]
- [UNG06]** UNIVERSIDAD DE GUANAJUATO. Glosario de Términos Relacionados con la Calidad en la UG, [Documento Electrónico] <  
<  
<http://usic13.ugto.mx/calidad/archivos/Glosario.pdf>.> [Consulta: 13 de Junio de 2006]
- [GRO03]** GREEVY, Orla. The Achievement and Validation of Evolution-Oriented Software Systems, white paper [Documento Electrónico], Marzo 2003. <

<http://www.iam.unibe.ch/~greevy/download/proposal.pdf>>

[Consulta: 25 de Abril de 2006]

- [ISI05]** ISO/IEC 15504 [en línea], Enero 2005. Disponible en web: <[http://www.isospice.typepad.com/isospice\\_is15504](http://www.isospice.typepad.com/isospice_is15504)>. [Consulta: 8 de Abril de 2006]
- [DUS03]** DUNCAN, Scott P., Making sense of ISO 15504 (and SPICE) , white paper [Documento Electrónico], Mayo 2003 <[http://www.westfallteam.com/Papers/Making\\_Sense\\_of\\_15504.pdf](http://www.westfallteam.com/Papers/Making_Sense_of_15504.pdf)>. [Consulta: 9 de Abril de 2006]
- [PRA06]** PROCESS ATTRIBUTES [en línea]. Disponible en web: <[http://isospice.typepad.com/isospice\\_assessment\\_model/2005/01/process\\_attribu.html](http://isospice.typepad.com/isospice_assessment_model/2005/01/process_attribu.html)> [Consulta: 9 de Abril de 2006]
- [TEN06]** Terrazas, Nelson. Valoración y Mejora de Procesos Software [Documento Electrónico], Febrero 2006 <<http://www.tevasoft.com/ISAvanzada2006/ValProcSoftwareV2.pdf>>. [Consulta: 12 de Abril de 2006]
- [PIF06]** PINO, F.J. *et al.* Adaptación de las normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la evaluación de la madurez de procesos software en países en desarrollo, white paper [Documento Electrónico], Abril 2006 <[http://www.ewh.ieee.org/reg/9/etrans/vol4issue2April2006/4TLA2\\_4Pino.pdf](http://www.ewh.ieee.org/reg/9/etrans/vol4issue2April2006/4TLA2_4Pino.pdf)>. [Consulta: 12 de Abril de 2006]
- [COC01]** CONCEPT OF OPERATIONS FOR THE CMMI [en línea], Enero 2001. Disponible en web: <<http://www.sei.cmu.edu/cmmi/background/conops.html>> [Consulta: 1 de Junio de 2006]
- [SEI97]** SEI, IPD-CMM - Integrated Product Development. CMU/SEI-MM-97-001, 1997.

- [COF02]** COOPER, Jack y FISHER, Matt. Software Acquisition Capability Maturity Model. Technical Report (SA-CMM). CMU/SEI-2002-TR-010, 2002.
- [UNC05]** UNIVERSIDAD DEL CAUCA. Hacia una Línea de Procesos Ágiles Agile SPsL [Documento Electrónico], Mayo 2005. <<http://www.dcc.uchile.cl/~cecilia/papers/AgileSPsL.pdf>> [Consulta: 12 de Junio de 2006]
- [WIC06]** WIKIPEDIA. CMMI [en línea]. Disponible en web: <<http://es.wikipedia.org/wiki/CMMI>> [Consulta: 15 de Junio de 2006]
- [CKS03]** CHRISSIS, Beth, KONRAD, Mike y SHRUM, Sandy. CMMI: Guidelines for Process Integration and Product Improvement. Addison-Wesley, 2003. 663 p.
- [GAL06]** GALLAGHER, Brian. Use of CMMI in Acquisition Environments. En: SYSTEMS & SOFTWARE TECHNOLOGY. (1º 2006 : Pittsbugh). Systems & Software Technology Conference. Pittsbugh: Carnegie Mellon University, 2006. 35 p.
- [SEI05]** SOFTWARE ENGINEERING INSTITUTE. CMMI Acquisition Module (CMMI-AM), Version 1.1. CMU/SEI-2005-TR-011, 2005.
- [SVE01]** SVENNBERG, Daniel. SOFTWARE ACQUISITION MANAGEMENT GUIDELINES. Suecia, Linköping, 2001. Trabajo de grado. Linköping University. The Department of Computer and Information Science.
- [HSC97]** HAIMES, Yacov, SCHOOFF, Richard M. y CHITTISTER, Clyde. A Holistic Management Framework for Software Acquisition [Documento Electrónico], Defense Systems Management College, 1997, p. 55-86. <<http://www.dau.mil/pubs/arg/97arg/schoof.pdf>>. [Consulta: 11 de Junio de 2006]

- [GAM99]** GALLIVAN, Michael J., Analyzing IT Outsourcing Relationships as Alliances among Multiple Clients and Vendors, white paper [Documento Electrónico], 1999. <<http://csdl2.computer.org/comp/proceedings/hicss/1999/0001/07/00017066.PDF>>
- [FAR06]** Failure Rate: Statistics over IT projects failure rate [en línea]. Disponible en web: <[http://www.it-cortex.com/Stat\\_Failure\\_Rate.htm](http://www.it-cortex.com/Stat_Failure_Rate.htm)> [Consulta: 30 de Mayo de 2006]
- [STG95]** THE STANDISH GROUP REPORT. The CHAOS Report, white paper [Documento Electrónico], 1995. <[http://www.projectsart.co.uk/docs/chaos\\_report.pdf](http://www.projectsart.co.uk/docs/chaos_report.pdf)>
- [SAA98]** SALWIN, A. E., The Road to Successful ITS Software Acquisition: Software Acquisition Process Reference Guide [Documento Electrónico]. Washington: Federal Highway Administration, 1998. 2 v. <[http://199.79.179.101/tfhrc/safety/pubs/its/architecture/rdsuccess\\_vol2.pdf](http://199.79.179.101/tfhrc/safety/pubs/its/architecture/rdsuccess_vol2.pdf)>
- [BRO98]** BROWN, Norm *et al.* The Program Manager's Guide to Software Acquisition Best Practices version 2.2., Arlington: Software Program Managers Network, 1998
- [SOS97]** SOMMERVILLE, Ian y SAWYER, Peter. Requirements Engineering: A Good Practice Guide. John Wiley and Sons, 1997. 404 p.
- [ROY70]** ROYCE, Winston. Managing the Development of Large Software Systems: Concepts and Techniques. Proceedings of WESCON, 1970.

- [JBR00]** JACOBSON, Ivar, BOOCH, Grady y RUMBAUGH, James. El Proceso Unificado de Desarrollo de Software. Madrid: Addison-Wesley, 2000. 438 p.
- [OSG96]** OSKARSSON, Östen y GLASS, Robert L., An ISO 9000 Approach to Building Quality Software. Upper Saddle River: Prentice-Hall, Inc., 1996.
- [PER04]** PERALTA, Mario. ESTIMACIÓN DEL ESFUERZO BASADA EN CASOS DE USO, white paper [Documento Electrónico], 2004. <<http://www.itba.edu.ar/capis/rtis/rtis-6-1/estimaci%F3n-del-esfuerzo-basada-en-casos-de-usos.pdf>>. [Consulta: 16 de Abril de 2006]
- [RUP03]** IBM Rational Software Corporation. Rational Unified Process, version 2003.06.12 [CD-ROM]. Rational Software: Cupertino, California, 2003
- [GAB01]** GALLAGHER, Brian y BROWNSWORD, Lisa. The Rational Unified Process and The Capability Maturity Model - Integrated Systems/Software Engineering [Documento Electrónico]. Pittsburgh: Carnegie Mellon, 2001 <<http://www.processwave.net/Downloads/Environment/CMMI&RUP/ESEPG2001%20.pdf>>. [Consulta: 30 de Mayo de 2006]
- [DUB00]** DURÁN, Amador y BERNÁRDEZ, Beatriz. Metodología para la Elicitación de Requisitos de Sistemas Software, versión 2.1 [Documento Electrónico]. Sevilla: Universidad de Sevilla, 2000 <<http://www.lsi.us.es/~informes/lsi-2000-10.pdf>>
- [WEI98]** WEIDENHAPUT, K., Scenarios in System Development: Current Practice. IEEE Software, 1998. p. 34–45.
- [RZF97]** RAGHAVAN, S., ZELESNIK, G., y FORD, G., Lecture Notes on Requirements Elicitation. CMU/SEI-94-EM-10 / ADA278536, 1997.

- [ISO91]** ISO/IEC 9126. Software Product Evaluation – Quality Characteristics and Guidelines for their use. 1991.
- [GIC95]** Government/Industry Committee. PARAMETRIC COST ESTIMATING HANDBOOK [Documento Electrónico]. Steering Committee, 1995. <<http://www1.jsc.nasa.gov/bu2/PCEH/PCEH-5.DOC>> [Consulta: 15 de Junio de 2006]
- [ASN97]** ASNER, Michael. The Request for Proposal Handbook: A Sourcebook of Guidelines. Sacramento, CA: Government Technology Press, 1997.
- [HUW06]** HUNT, Theresa y WESTFALL, Linda. Software Acquisition & Supplier Management: Part 1 – Product Definition & Supplier Selection, white paper [Documento Electrónico]. <[http://www.westfallteam.com/Papers/Software\\_Acquisition\\_1.pdf](http://www.westfallteam.com/Papers/Software_Acquisition_1.pdf)> [Consulta: 15 de Junio de 2006]
- [PRE00]** PRESSMAN, Roger S., Software Engineering: A Practitioner's Approach. McGrawHill, 2000. Quinta Edición.
- [WII06]** WIKIPEDIA. Installation Testing [en línea]. Disponible en web: [http://en.wikipedia.org/wiki/Installation\\_testing](http://en.wikipedia.org/wiki/Installation_testing) [Consulta: 1 de Junio de 2006]
- [SUT06]** Support Testing [en línea]. Disponible en web: <<http://fox.wikis.com/wc.dll?Wiki~SupportTesting>> [Consulta: 1 de Junio de 2006]
- [JBR99]** JACOBSON, Ivar, BOOCH, Grady y RUMBAUGH, James. El Lenguaje Unificado de Modelado, Manual de Referencia. Madrid: Addison-Wesley, 1999. 432 p.
- [GLM06]** GLINZ, Martin. Improving the Quality of Requirements with Scenarios, white paper [Documento Electrónico].

<http://www.ifi.unizh.ch/groups/req/ftp/papers/2WCSQ.pdf>

[Consulta: 15 de Junio de 2006]

- [KEN06]** Kerth, Norman L., An Approach to Post Morta, Post Parta, and Post Project Reviews, white paper [Documento Electrónico]. Beaverton: Elite Systems <<http://c2.com/doc/ppm.pdf>> [Consulta: 26 de Junio de 2006].