



Técnico en
< DESARROLLO DE SOFTWARE >

***Análisis de Requisitos del
Sistema y del Software***

(CC BY-NC-ND 4.0)
International

Attribution-NonCommercial-NoDerivatives 4.0



Atribución

Usted debe reconocer el crédito de una obra de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace.



No Comercial

Usted no puede hacer uso del material con fines comerciales.



Sin obra derivada

Si usted mezcla, transforma o crea un nuevo material a partir de esta obra, no puede distribuir el material modificado.

No hay restricciones adicionales - Usted no puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otros hacer cualquier uso permitido por la licencia.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



Análisis de Requisitos del Sistema y del Software

Unidad I

Procesos de la ingeniería de requerimientos

1. Definición de Requerimientos y de Análisis de requerimientos

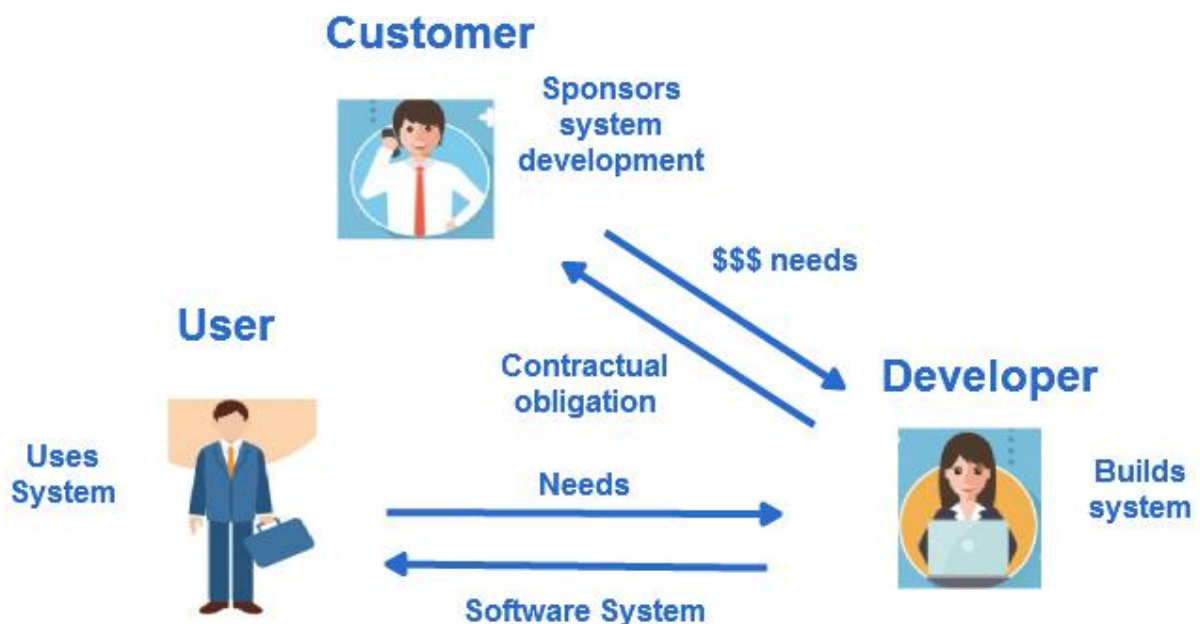


Figura 1.1 Participantes en el desarrollo de un software.

Requerimientos: Los requerimientos especifican qué es lo que el sistema debe hacer (sus funciones) y sus propiedades esenciales y deseables.

La captura de los requerimientos tiene como objetivo principal la comprensión de lo que los clientes y los usuarios esperan que haga el sistema. Un requerimiento expresa el propósito del sistema sin considerar como se va a implantar. En otras palabras, los requerimientos identifican el **qué** del sistema, mientras que el diseño establece el **cómo** del sistema.

La captura y el análisis de los requerimientos del sistema es una de las fases más importantes para que el proyecto tenga éxito. Como regla de modo empírico, el costo de reparar un error se incrementa en un factor de diez de una fase de desarrollo a la siguiente, por lo tanto la preparación de una especificación adecuada de requerimientos reduce los costos y el riesgo general asociado con el desarrollo [Norris & Rigby, 1994].

Análisis de requerimientos: Es el conjunto de técnicas y procedimientos que nos permiten conocer los elementos necesarios para definir un proyecto de software.

Es una tarea de ingeniería del software que permite especificar las características operacionales del software, indicar la interfaz del software con otros elementos del sistema y establecer las restricciones que debe cumplir el software.

La especificación de requerimientos suministra al técnico y al cliente, los medios para valorar el cumplimiento de resultados, procedimientos y datos, una vez que se haya construido.

La tarea de análisis de los requerimientos es un proceso de descubrimiento y refinamiento, el cliente y el desarrollador tienen un papel activo en la ingeniería de requerimientos de software.

El cliente intenta plantear un sistema que en muchas ocasiones es confuso para él, sin embargo, es necesario que describa los datos, que especifique las funciones y el comportamiento del sistema que desea. El objetivo es que el desarrollador actúe como un negociador, un interrogador, un consultor, o sea, como persona que consulta y propone para resolver las necesidades del cliente.

El análisis de requerimientos proporciona una vía para que los clientes y los desarrolladores lleguen a un acuerdo sobre lo que debe hacer el sistema. La especificación, producto de este análisis proporciona las pautas a seguir a los diseñadores del sistema.

2. Tipos de requerimientos.

Según el estándar internacional de Especificación de Requerimientos IEEE830, los documentos de definición y especificación de requerimientos deben contemplar los siguientes aspectos resumidos por [Pfleeger, 2002] como se indica a continuación:

Ambiente físico

- ¿Dónde está el equipo que el sistema necesita para funcionar?
- ¿Existe una localización o varias?
- ¿Hay restricciones ambientales como temperatura, humedad o interferencia magnética?

Interfaces

- ¿La entrada proviene de uno o más sistemas?
- ¿La salida va a uno o más sistemas?

- ¿Existe una manera preestablecida en que deben formatearse los datos?

Usuarios y factores humanos

- ¿Quién usará el sistema?
- ¿Habrá varios tipos de usuario?
- ¿Cuál es el nivel de habilidad de cada tipo de usuario?
- ¿Qué clase de entrenamiento requerirá cada tipo de usuario?
- ¿Cuán fácil le será al usuario comprender y utilizar el sistema?
- ¿Cuán difícil le resultará al usuario hacer uso indebido del sistema?

Funcionalidad

- ¿Qué hará el sistema?
- ¿Cuándo lo hará?
- ¿Existen varios modos de operación?
- ¿Cómo y cuándo puede cambiarse o mejorarse un sistema?
- ¿Existen restricciones de la velocidad de ejecución, tiempo de respuesta o rendimiento?

Documentación

- ¿Cuánta documentación se requiere?
- ¿Debe estar en línea, en papel o en ambos?
- ¿A qué audiencia está orientado cada tipo de información?

Datos

- ¿Cuál será el formato de los datos, tanto para la entrada como para la salida?
- ¿Cuán a menudo serán recibidos o enviados?
- ¿Cuán exactos deben ser?
- ¿Con qué grado de precisión deben hacerse los cálculos?
- ¿Cuántos datos fluyen a través del sistema?
- ¿Debe retenerse algún dato por algún período de tiempo?

Recursos

- ¿Qué recursos materiales, personales o de otro tipo se requieren para construir, utilizar y mantener el sistema?
- ¿Qué habilidades deben tener los desarrolladores?
- ¿Cuánto espacio físico será ocupado por el sistema?
- ¿Cuáles son los requerimientos de energía, calefacción o acondicionamiento de aire?
- ¿Existe un cronograma prescrito para el desarrollo?
- ¿Existe un límite sobre la cantidad de dinero a gastar en el desarrollo o en hardware y software?

Seguridad

- ¿Debe controlarse el acceso al sistema o a la información?
- ¿Cómo se podrán aislar los datos de un usuario de los de otros?
- ¿Cómo podrán aislarse los programas de usuario de los otros programas y del sistema operativo?
- ¿Con qué frecuencia deben hacerse copias de respaldo?

- ¿Las copias de respaldo deben almacenarse en un lugar diferente?
- ¿Deben tomarse precauciones contra el fuego, el daño provocado por agua o el robo?

Aseguramiento de la calidad

- ¿Cuáles son los requerimientos para la confiabilidad, disponibilidad, facilidad de mantenimiento, seguridad y demás atributos de calidad?
- ¿Cómo deben demostrarse las características del sistema a terceros?
- ¿El sistema debe detectar y aislar defectos?
- ¿Cuál es el promedio de tiempo prescrito entre fallas?
- ¿Existe un tiempo máximo permitido para la recuperación del sistema después de una falla?
- ¿El mantenimiento corregirá los errores, o incluirá también el mejoramiento del sistema?
- ¿Qué medidas de eficiencia se aplicarán al uso de recursos y al tiempo de respuesta?
- ¿Cuán fácil debe ser mover el sistema de una ubicación a otra o de un tipo de computadora a otro?

3. Características de los requerimientos.

Los requerimientos permiten que los desarrolladores expliquen cómo han entendido lo que el cliente pretende del sistema. También, indican a los diseñadores qué funcionalidad y que características va a tener el sistema resultante. Y además, indican al equipo de pruebas qué demostraciones llevar a cabo para convencer al cliente de que el sistema que se le entrega es lo que solicitó. Las características de los requerimientos mencionados en el estándar IEEE830 los explica [Pfleeger, 2002] como sigue:

Deben ser correctos.

Tanto el cliente como el desarrollador deben revisarlos para asegurar que no tienen errores.

Deben ser consistentes.

Dos requerimientos son inconsistentes cuando es imposible satisfacerlos simultáneamente.

Deben estar completos.

El conjunto de requerimientos está completo si todos los estados posibles, cambios de estado, entradas, productos y restricciones están descritos en alguno de los requerimientos.

Deben ser realistas.

Todos los requerimientos deben ser revisados para asegurar que son posibles.

¿Cada requerimiento describe algo que es necesario para el cliente?

Los requerimientos deben ser revisados para conservar sólo aquellos que inciden directamente en la resolución del problema del cliente.

Deben ser verificables.

Se deben poder preparar pruebas que demuestren que se han cumplido los requerimientos.

Deben ser rastreables.

¿Se puede rastrear cada función del sistema hasta el conjunto de requerimientos que la establece?

4. Métodos generales de entrevistas.

La entrevista es una forma de recoger información de otra persona a través de una comunicación interpersonal que se lleva a cabo por medio de una conversación estructurada, [Braude, 2003] distingue las siguientes fases:

- *Preparación:* El entrevistador debe documentarse e investigar la situación de la organización, analizando los documentos de la empresa disponible. Hay que intentar minimizar el número de entrevistados, hay que considerar las entrevistas de cortesía, analizar el perfil de los entrevistados, definir el objetivo y el contenido de la entrevista, planificar el lugar y la hora en la que se va a desarrollar la entrevista es conveniente realizarla en un lugar confortable.

Algunos proponen enviar previamente el entrevistado un cuestionario y un pequeño documento de introducción al proyecto de desarrollo.

- *Realización:* Hay tres fases:

1. **Apertura:** Presentarse e informar al entrevistado sobre la razón de la entrevista.
2. **Desarrollo:** Cumplir las reglas del protocolo, hay que llegar a un acuerdo sobre cómo se va a registrar la información obtenida.

3. **Terminación:** Se termina recapitulando la entrevista agradeciendo el esfuerzo y dejando abierta la posibilidad de volver a contactar para aclarar conceptos o bien citándole para otra entrevista.

- *Análisis:* Consiste en leer las notas, pasarlas en limpio, reorganizar la información, contrastarlas con otras entrevistas o fuentes de información, evaluar cómo ha ido la entrevista.

Las entrevistas con los involucrados con el sistema son parte de la mayoría de los procesos de la ingeniería de requerimientos. En estas entrevistas, el equipo de la ingeniería de requerimientos hace preguntas sobre el sistema que utilizan y sobre el sistema a desarrollar.

Los requerimientos provienen de las respuestas a estas preguntas.

Las entrevistas pueden ser de dos tipos:

1. *Entrevistas cerradas:* donde los entrevistados responden a un conjunto predefinido de preguntas.

2. *Entrevistas abiertas:* donde no hay un programa predefinido. El equipo de la ingeniería de requerimientos examina una serie de cuestiones con los involucrados con el sistema y, por lo tanto, desarrolla una mejor comprensión de sus necesidades.

En la práctica, las entrevistas son una mezcla de estos dos tipos. Las respuestas a algunas preguntas pueden conducir a otras cuestiones que se discuten de una forma menos estructurada. Las discusiones completamente abiertas rara vez salen bien; la

mayoría de las entrevistas requieren algunas preguntas para empezar y para mantener la entrevista centrada en el sistema a desarrollar.

Las entrevistas sirven para obtener una comprensión general de lo que hacen los futuros usuarios del sistema, cómo podrían interactuar con el sistema y las dificultades a las que se enfrentan con los sistemas actuales. A la gente le gusta hablar de su trabajo y normalmente se alegran de verse implicados en las entrevistas. Sin embargo, no son de tanta utilidad para la comprensión de los requerimientos del dominio de la aplicación, tampoco son una técnica eficaz para obtener conocimiento sobre los requerimientos y restricciones organizacionales debido a que existen sutiles poderes e influencias entre los involucrados en el sistema. En general, la mayoría de la gente es reacia a discutir cuestiones políticas y organizacionales que pueden influir en los requerimientos. Por otra parte, hay que destacar que para la mayoría de las personas, la entrevista es un compromiso adicional sobre su cargada lista de trabajos pendientes. Algunos autores proponen mandar previamente un cuestionario que debe llenar el entrevistado y un pequeño documento de introducción al proyecto de desarrollo. El cuestionario permite que el entrevistado conozca los temas que se van a tratar y pueda conseguir con anticipación información que no tenga a disposición inmediata.

Los buenos entrevistadores poseen dos características importantes:

- a) No tienen prejuicios, evitan ideas preconcebidas sobre los requerimientos y están dispuestos a escuchar a los entrevistados. Si el entrevistado propone requerimientos sorprendentes, están dispuestos a cambiar su opinión del sistema.

- b) Ayudan al entrevistado a empezar las discusiones con una pregunta, una propuesta de requerimientos o sugiriendo trabajar juntos en un prototipo del sistema.

Preguntar al cliente por lo que quiere de manera general normalmente no proporciona información útil. Para la mayoría de la gente es mucho más fácil hablar de algo en particular que en términos generales.

Las entrevistas, son una técnica general para obtener información. Se pueden complementar las entrevistas individuales con entrevistas en grupo o grupos de discusión. Las ventajas de utilizar grupos de discusión es que los usuarios se estimulan entre sí para proporcionar información y pueden terminar discutiendo diferentes formas que han desarrollado para utilizar los sistemas.

Más adelante, en la semana VII, se expone el “Desarrollo de Conjunto de Aplicaciones” (JAD).

La información de las entrevistas complementa otras informaciones sobre el sistema además de los documentos, observaciones de los usuarios, etc. Las entrevistas tienden a omitir información esencial, por lo que deberían ser usadas junto con otras técnicas de obtención de requerimientos.

5. Recomendaciones generales

Es común que haya varios interesados en dar su opinión, lo primero es decidir a quién entrevistar. En lugar de intentar que se dedique el mismo tiempo a todos, lo cual puede resultar en requerimientos contradictorios y esfuerzo desperdiciado, [Braude, 2003] recomienda seleccionar uno o quizá dos individuos principales, entrevistarlos y después

solicitar comentarios de otros interesados clave. Es preferible que haya dos entrevistadores en cada sesión, pues un entrevistador típico tiende a perder puntos. Grabar la entrevista suele ayudar, pero debe pedirse permiso de antemano. El objetivo principal es minimizar el número necesario de personas a entrevistar para obtener una visión lo más completa posible sobre el sistema a desarrollar, sin embargo, hay que considerar también las entrevistas de “cortesía”, por ejemplo, al jefe de la unidad que se analiza o de quien depende el sistema, quien aportará una visión estratégica que podría ser de interés, pero sobre todo, de quien se persigue obtener el permiso y el apoyo para poder entrevistar al resto del personal. Hay que tener en cuenta que se eliminan muchas dificultades para entrevistar a los empleados si su jefe avala la iniciativa [Piattini et al., 2004].

Una manera de manejar las entrevistas:

Antes de la entrevista.

1. Enumerar y dar prioridad a los clientes que se entrevistarán.
2. Programar una entrevista con tiempos de inicio y terminación fijos.

En la entrevista

1. No ser pasivo, investigar y animar, persistir en entender *deseos* y explorar *necesidades*.
2. Examinar casos de uso, flujos de datos y/o diagramas de estado.
3. Tomar notas exhaustivas.

4. Programar una reunión de seguimiento.

Después de la entrevista.

1. Bosquejar la especificación de los requerimientos.
2. Enviar correos electrónicos a los clientes para obtener sus comentarios.

El entrevistado puede presentar:

- ☹ Pasividad, inhibición
- ☹ No aceptación
- ☹ Rechazo
- ☹ Agresividad

El entrevistador debe poseer:

- 😊 Trato cordial
- 😊 Conocimiento de técnicas de comunicación
- 😊 Actitud para escuchar sin prejuicios
- 😊 Experiencia práctica
- 😊 Interés por el tema

No basta con hacer preguntas, es importante la forma en que se plantea la conversación y la relación que se establece.

6. Importancia de la definición formal de requerimientos.

El análisis y especificación de requerimientos puede parecer una tarea relativamente sencilla, pero las apariencias engañan. Puesto que el contenido de comunicación es muy alto, abundan los cambios por mala interpretación o falta de información.

El dilema con el que se enfrenta un ingeniero de software puede ser comprendido repitiendo la sentencia de un cliente anónimo: "Sé que crees que comprendes lo que piensas que he dicho, pero no estoy seguro de que entendiste lo que yo quise decir".

En la *tabla 1.1* [Pfleeger, 2002] ilustra el conflicto que encontró (Scharer, 1990) cuando los desarrolladores y los usuarios se limitan a ver el problema desde su particular punto de vista sin tomar en cuenta la situación del otro.

Como ven los desarrolladores a los usuarios.	Como ven los usuarios a los desarrolladores.
Los usuarios no saben lo que quieren.	Los desarrolladores no comprenden las necesidades operacionales.
Los usuarios no pueden articular lo que quieren.	Los desarrolladores ponen demasiado énfasis en la técnica.
Los usuarios tienen muchas necesidades motivadas políticamente.	Los desarrolladores pretenden decirnos como hacer nuestro trabajo.

Los usuarios lo quieren todo bien y ahora.	Los desarrolladores no pueden traducir nuestras necesidades claramente establecidas a un sistema exitoso.
Los usuarios son incapaces de priorizar sus necesidades.	Los desarrolladores dicen “no” todo el tiempo.
Los usuarios rehúsan tomar responsabilidades por el sistema.	Los desarrolladores siempre están por encima del presupuesto.
Los usuarios son incapaces de proporcionar un enunciado utilizable de las necesidades.	Los desarrolladores siempre están atrasados.
Los usuarios no están comprometidos con los proyectos de desarrollo de sistemas.	Los desarrolladores piden a los usuarios tiempo y esfuerzo, aún en detrimento de sus obligaciones primarias importantes.
Los usuarios no tienen voluntad de colaborar.	Los desarrolladores establecen estándares no realistas para la definición de los requerimientos.
Los usuarios no pueden mantener el cronograma.	Los desarrolladores son incapaces de responder rápidamente a los legítimos cambios de las necesidades.

Tabla 1.1: Usuarios y Desarrolladores: como se ven el uno al otro (Scharer 1990).

En realidad existen muchos contribuyentes al conjunto de los requerimientos. Cada uno tiene una visión particular del sistema de cómo debe funcionar, a menudo estas visiones

son conflictivas. Una de las muchas habilidades de un analista de requerimientos es la capacidad para comprender cada punto de vista y capturar los requerimientos de una manera que refleje los intereses de cada participante.

7. Procesos de la Ingeniería de Requerimientos.

En cursos previos de Ingeniería de Software se aprende que existen diferentes modelos para desarrollar sistemas de software (cascada, evolutivos, etc.) y la obtención de requerimientos se ve como un subproceso de este desarrollo. Sin embargo, visto por separado, el Análisis de Requerimientos es todo un proceso al cual se llama “Ingeniería de Requerimientos” cuya meta es crear y mantener un documento de requerimientos del sistema. Este proceso general consta de cuatro subprocesos:

El estudio de viabilidad, que evalúa si el sistema es útil para el negocio.

- Obtención y análisis de requerimientos.
- Especificación de requerimientos: transformación de los requerimientos en formularios estándar.
- Validación: verificar que los requerimientos realmente definen el sistema que quiere el cliente.

En la *figura 2.1* se muestra el proceso de ingeniería de requerimientos.

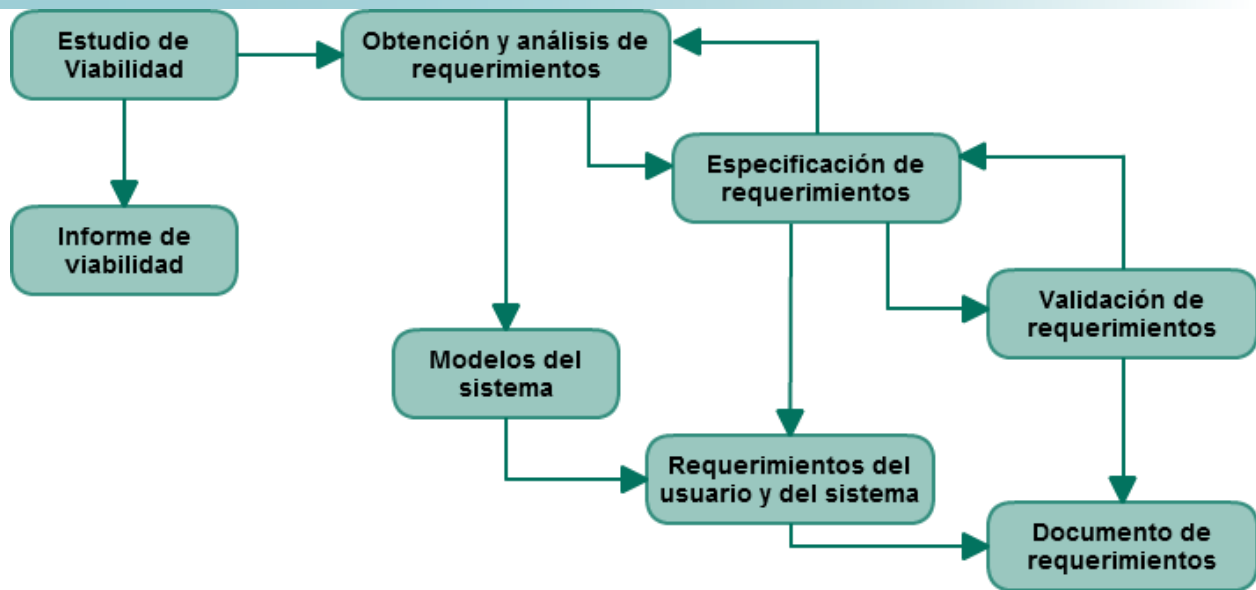


Figura 1.2: Proceso de ingeniería de requerimientos

7.1. Estudios de viabilidad.

Se define el estudio de viabilidad como un estudio corto y orientado a resolver las siguientes preguntas:

- 1.- ¿El sistema contribuye a los objetivos generales de la organización o empresa?
- 2.- ¿El sistema se puede implantar utilizando tecnología actual dentro de las restricciones de tiempo y presupuesto?
- 3.- ¿El sistema puede integrarse a otros sistemas existentes en la empresa?

Para ayudar a responder las preguntas del estudio de viabilidad, se tienen algunos ejemplos de preguntas posibles:

- ¿Cómo se las arreglaría la organización o empresa si no se implantara el sistema?

- ¿Cuáles son los problemas con los procesos actuales y como ayudaría un sistema nuevo a aliviarlos?
- ¿Cuál es la contribución directa que hará el sistema a los objetivos y requerimientos del negocio?
- ¿Se puede obtener y transferir la información a otros sistemas de la organización?
- ¿El sistema requiere tecnología que no se ha utilizado previamente en la organización?
- ¿A que debe ayudar el sistema y a qué no necesita ayudar?

El estudio de viabilidad no debe requerir más de dos o tres semanas. El resultado de este estudio es un informe que recomiende si vale o no la pena seguir con la ingeniería de requerimientos y el proceso de desarrollo del sistema. En el informe se pueden proponer cambios en el alcance, el presupuesto o sugerir requerimientos adicionales de alto nivel.

7.2. Obtención y análisis de requerimientos.

La siguiente etapa del proceso de ingeniería de requerimientos es la obtención y análisis de requerimientos. En esta actividad, los ingenieros de software trabajan con los clientes y los usuarios finales del sistema para determinar el dominio de la aplicación, qué servicios debe proporcionar el sistema, el rendimiento requerido del sistema, las restricciones hardware, etcétera.

Se presenta el modelo de la *figura 1.3* para mostrar que los requerimientos pueden extraerse de muchas maneras, sugiere ser creativos en la forma de averiguar qué es lo que los clientes quieren, y se propone:

- Revisar la situación actual.
- Trabajar en el ámbito del usuario para comprender el contexto, los problemas y las relaciones.
- Entrevistar a los usuarios actuales y potenciales.
- Realizar un video para mostrar cómo podría funcionar el nuevo sistema.
- Investigar en documentos existentes.
- Conducir tormentas de ideas con los usuarios actuales y potenciales.
- Observar las estructuras y los patrones.

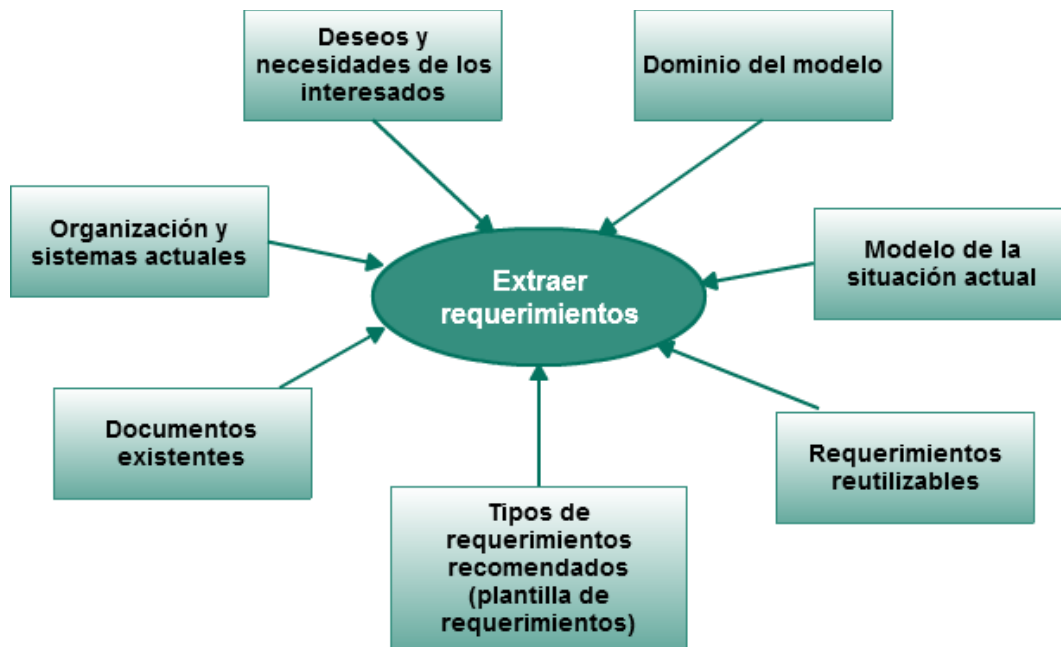


Figura 1.3: Posibles fuentes requerimientos (Robertson y Robertson, 1999).

La obtención y análisis de requerimientos pueden afectar a varias personas de la organización. El término *stakeholder* (sin traducción al español) se utiliza en la Ingeniería de Software para referirse a cualquier persona o grupo que se verá afectado por el sistema, directa o indirectamente.

Obtener y comprender los requerimientos de los stakeholders es difícil por varias razones:

- Los stakeholders a menudo no conocen lo que desean obtener del sistema informático excepto en términos muy generales. Pueden hacer demandas irreales o resultarles difícil expresar lo que quieren que haga el sistema.
- Los ingenieros de requerimientos, sin experiencia en el dominio del cliente, deben comprender los requerimientos que los stakeholders expresan con sus propios términos y con un conocimiento implícito de su trabajo.
- Diferentes stakeholders tienen requerimientos distintos. Es necesario descubrir las concordancias y los conflictos entre éstos.
- Los factores políticos pueden influir en los requerimientos del sistema. Por ejemplo, los directivos pueden solicitar requerimientos específicos del sistema que incrementarán su influencia en la organización.
- Pueden emerger nuevos requerimientos de nuevos stakeholders que no habían sido consultados previamente.

El *descubrimiento de requerimientos* es el proceso de recoger información sobre el sistema propuesto y los existentes extrayendo esta información del usuario y del sistema. Las fuentes de información durante la fase del descubrimiento de requerimientos incluyen la documentación, los stakeholders del sistema y la especificación de sistemas similares. Las técnicas de descubrimiento de requerimientos, son varias, pueden usarse entrevistas, escenarios, prototipos y etnografía.

La figura 1.4 explica por qué el proceso de los requerimientos es crítico para el buen desarrollo de software.

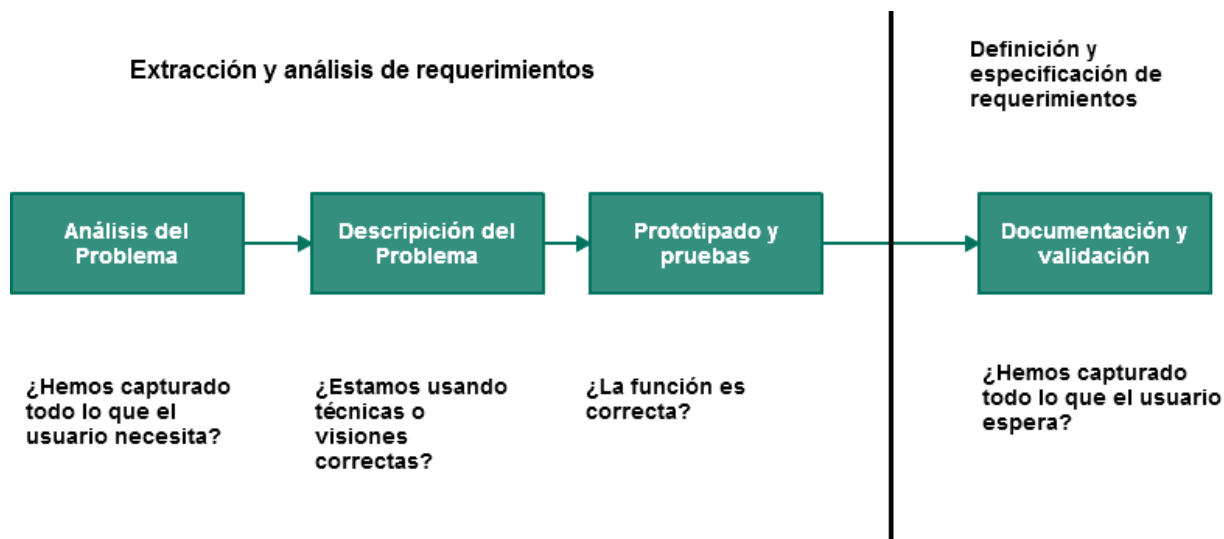


Figura 1.4: El proceso de determinación de los requerimientos (Fleeger, 2002).

7.3. Validación de requerimientos.

La validación de requerimientos sirve para demostrar que éstos realmente definen el sistema que el cliente desea. Asegura que los requerimientos están completos, son exactos y consistentes. Debe garantizar que lo descrito es lo que el cliente pretende ver en el producto final. Esta validación es importante porque la detección de errores durante el proceso de análisis de requerimientos reduce mucho los costos. Si se detecta un cambio en los requerimientos una vez que el sistema está hecho, los costos son muy altos, ya que significa volver a cambiar el diseño, modificar la implementación del sistema y probarlo nuevamente. Se plantea que las verificaciones que deben llevarse a cabo durante el proceso de validación, son las siguientes:

Verificación de validez. El análisis puede identificar que se requieren funciones adicionales o diferentes a las que pidieron los stakeholders.

- *Verificación de consistencia.* No debe haber restricciones o descripciones contradictorias en el sistema.
- *Verificación de completitud.* El documento de requerimientos debe incluir requerimientos que definan todas las funciones y restricciones propuestas por el usuario del sistema.
- *Verificación de realismo.* Asegurar que los requerimientos pueden cumplirse teniendo en cuenta la tecnología existente, el presupuesto y el tiempo disponible.
- *Verificabilidad.* Para reducir la posibilidad de discusiones con el cliente, los requerimientos del sistema siempre deben redactarse de tal forma que sean verificables. Esto significa que se debe poder escribir un conjunto de pruebas que demuestren que el sistema a entregar cumple cada uno de los requerimientos especificados.

Existen varias técnicas de validación de requerimientos, estas son: *revisiones de requerimientos, construcción de prototipos y generación de casos de prueba.*

Una *revisión de requerimientos* es un proceso manual en la que intervienen tanto el cliente como personal involucrado en el desarrollo del sistema, ésta puede ser formal o informal, y tiene el fin de verificar que el documento de requerimientos no presente anomalías ni omisiones. En una revisión formal, los revisores deben tomar en cuenta:

- Que el requerimiento se pueda verificar de modo realista.

- Que las personas que adquieren el sistema o los usuarios finales comprendan correctamente el requerimiento.
- ¿Qué tan adaptable es el requerimiento? Es decir, ¿puede cambiarse el requerimiento sin causar efectos de gran escala en los otros requerimientos del sistema?

La *construcción de prototipos* consiste en mostrar un modelo ejecutable del sistema a los usuarios finales y a los clientes, así éstos pueden experimentar con el modelo para ver si cumple con sus necesidades reales.

Los requerimientos deben poder probarse, es por esto que debe hacerse una *generación de casos de prueba*. Si una prueba es difícil o imposible de diseñar, normalmente significa que los requerimientos serán difíciles de implantar y deberían ser considerados nuevamente.

En resumen, la validación pretende asegurar que los requerimientos satisfarán las necesidades del cliente.

7.4. Gestión de requerimientos.

(Manejo de los cambios de requerimientos durante la construcción).

En la práctica, en casi todos los sistemas los requerimientos cambian. Las personas involucradas desarrollan una mejor comprensión de lo que quieren que haga el software; la organización que compra el sistema cambia; se hacen modificaciones a los sistemas de hardware, software y al entorno organizacional. El proceso de organizar y llevar a cabo los cambios en los requerimientos se llama *gestión de requerimientos*.

El objetivo del analista es reconocer los elementos básicos de un sistema tal como lo percibe el usuario/cliente. El analista debe establecer contacto con el equipo técnico y de gestión del usuario/cliente y con la empresa que vaya a desarrollar el software. El gestor del programa puede servir como coordinador para facilitar el establecimiento de los caminos de comunicación.

Una vez que un sistema se ha instalado, inevitablemente surgen nuevos requerimientos. Es difícil para los usuarios y clientes del sistema anticipar qué efectos tendrá el sistema nuevo en la organización. Cuando los usuarios finales tienen experiencia con un sistema, descubren nuevas necesidades y prioridades.

Las personas que pagan por el sistema y los usuarios de éste, rara vez son la misma persona. Los clientes del sistema imponen requerimientos debido a las restricciones organizacionales y de presupuesto. Éstos pueden estar en conflicto con los requerimientos de los usuarios finales y, después de la entrega, pueden tener que añadirse nuevas características de apoyo al usuario para que el sistema cumpla con sus objetivos.

Otro motivo por el que cambian los requerimientos es que en ocasiones el entorno de negocios y técnico del sistema cambian después de la instalación. Puede ser que se introduzca un nuevo hardware, o puede ser que surja la necesidad de que el sistema interactúe con otros sistemas. También cambian las prioridades del negocio, las legislaciones y las regulaciones, y esto debe estar reflejado en sistema.

La gestión de requerimientos es el proceso de comprender y controlar los cambios en los requerimientos del sistema [Sommerville, 2005].

Es necesario mantenerse al tanto de los requerimientos particulares y mantener vínculos entre los requerimientos dependientes de forma que se pueda evaluar el impacto de los cambios en los requerimientos. El proceso de gestión de requerimientos debe empezar cuando esté disponible una versión preliminar del documento de requerimientos. Hay que establecer un proceso formal para implantar las propuestas de cambios y planear como se van a gestionar los requerimientos que cambian durante el proceso de obtención de requerimientos.

En la *figura 1.5* se muestra como se tiene una mejor comprensión de las necesidades de los usuarios conforme se va desarrollando la definición de los requerimientos. Esta nueva comprensión retroalimenta al usuario, quien puede proponer entonces un cambio en los requerimientos.

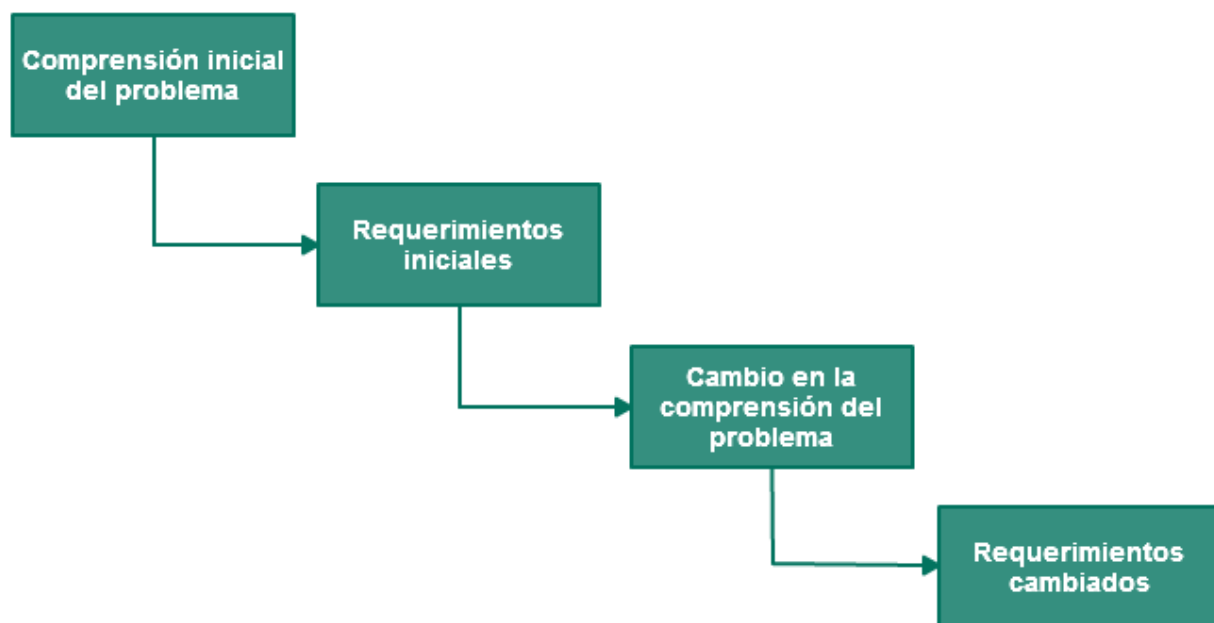


Figura 1.5: Evolución de los requerimientos (Sommerville 2005).

Desde una perspectiva evolutiva, los requerimientos se dividen en dos clases:

- *Requerimientos duraderos.* Son los relativamente estables, están relacionados directamente con la actividad principal de la organización.
- *Requerimientos volátiles.* Cambian durante el proceso de desarrollo o después de que éste se haya puesto en funcionamiento.

La *gestión del cambio en los requerimientos* (figura 2.5) se debe aplicar a todos los cambios propuestos en los requerimientos. La ventaja de utilizar un proceso formal para gestionar el cambio es que todos los cambios propuestos son tratados de forma consistente y que los cambios en el documento de requerimientos se hacen de forma controlada.

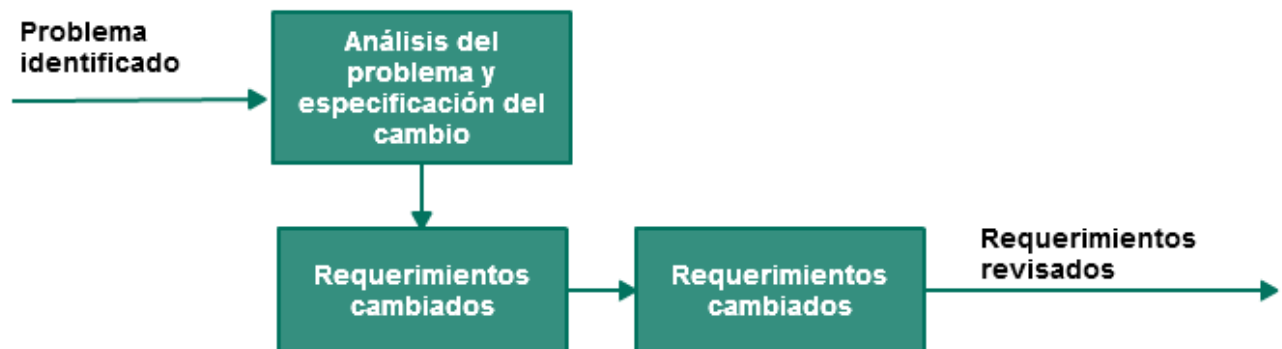


Figura 1.6: Gestión de cambios en los requerimientos (Sommerville 2005).

El proceso empieza con la identificación de un problema en los requerimientos o, algunas veces con una propuesta específica de cambio, dicha propuesta o el problema, se analiza para verificar que sea válido. Después se valora el efecto del cambio propuesto para calcular su costo. Este costo se estima tomando en cuenta las modificaciones al

documento de requerimientos, y si es el caso, al diseño e implementación del sistema.

Al finalizar esta etapa se toma una decisión sobre si se continúa con el cambio de requerimientos. En caso afirmativo, se lleva a cabo la implantación del cambio.

El documento de requerimientos debe organizarse haciendo sus secciones tan modulares como sea posible, de tal modo que se puedan hacer cambios en él sin tener que hacer grandes reorganizaciones o redactar nuevamente gran cantidad del mismo. Así se podrán cambiar o reemplazar secciones individuales sin afectar a otras partes del documento. Algunas veces pasa que se requiere de forma urgente un cambio en los requerimientos. Entonces surge la tentación de hacer ese cambio directamente en el sistema y luego modificar de forma retrospectiva el documento de requerimientos. Esto conduce casi inevitablemente a que la especificación de requerimientos y la implementación del sistema se desfasen. Una vez que se han hecho los cambios en el sistema, los del documento de requerimientos se pueden olvidar o se hacen de forma que no concuerdan con los cambios del sistema.

7.5. Ejemplo de un problema en los requerimientos (caso real).

“El desastre del cohete Ariane-5, fue provocado por la reutilización de una sección de código del Ariane-4. Nuseibeh (1997) analiza el problema desde el punto de vista de la reutilización de los requerimientos. Esto es, muchos ingenieros de software sienten que pueden obtener grandes beneficios reutilizando las especificaciones de requerimientos, (y también el diseño, el código y los casos de prueba relacionados) de sistemas anteriormente desarrollados. Las especificaciones candidatas son identificadas contemplando los requerimientos de funcionalidad o de comportamiento que son iguales

o similares, haciendo después las modificaciones donde fueran necesarias. En el caso del Ariane4, el sistema inercial de referencia (SRI) realizaba la mayoría de las funciones que necesitaba el Ariane-5.

Sin embargo, Nuseibeh señala que, aunque la funcionalidad que se necesitaba era similar a la del Ariane-4, había aspectos de Ariane-5 que eran significativamente diferentes. En particular la funcionalidad del módulo SRI que continuaba después del lanzamiento no se necesitaba para el Ariane-5. Es así que, si la validación de los requerimientos se hubiese realizado correctamente, los analistas habrían descubierto que las funciones activas después del lanzamiento no podían rastrearse hasta un requerimiento en la especificación o en la definición del Ariane-5. Es decir que la validación de los requerimientos podría haber jugado un papel crucial en la prevención de la destrucción del cohete.” [Pleeger, 2002].

Principales riesgos de la etapa de recolección de requerimientos.

El desarrollo de proyectos complejos de software lleva implícita la existencia de ciertos riesgos que si no se toman en cuenta y se analizan con cuidado, retrasarán considerablemente la entrega del producto o incluso podrían llegar a causar la cancelación del proyecto.

Un riesgo se puede definir de manera sencilla como:

- “la probabilidad de que una circunstancia adversa ocurra”
- “un problema potencial que puede ocurrir o no”

A continuación se muestran algunos elementos de riesgo identificados que pueden perjudicar la etapa de recolección de requerimientos:

- Los clientes no saben lo que quieren.
- Los clientes no quieren comprometerse a tener un conjunto de requerimientos escritos.
- Los clientes insisten en establecer nuevos requerimientos una vez que se han fijado la planificación y el coste.
- La comunicación con los clientes es lenta.
- Los clientes no participan en las revisiones o son incapaces de hacerlas.
- Los clientes no están preparados técnicamente.
- Los clientes no dejan realizar el trabajo a la gente.
- Los clientes no entienden el proceso de desarrollo de software.

Además se identifican los siguientes:

- No se comprende claramente el alcance del sistema.
- No se logran identificar con claridad los productos resultantes del proyecto.
- Ni los integrantes del equipo de desarrollo ni el cliente logran especificar de forma apropiada el área de aplicación.

Establecer buenas relaciones con los clientes permite identificar mejor los riesgos y controlarlos durante el desarrollo del proyecto.

Descargo de responsabilidad

La información contenida en este documento descargable en formato PDF o PPT es un reflejo del material virtual presentado en la versión online del curso. Por lo tanto, su contenido, gráficos, links de consulta, acotaciones y comentarios son responsabilidad exclusiva de su(s) respectivo(s) autor(es) por lo que su contenido no compromete al área de e-Learning del Departamento GES o al programa académico al que pertenece.

El área de e-Learning no asume ninguna responsabilidad por la actualidad, exactitud, obligaciones de derechos de autor, integridad o calidad de los contenidos proporcionados y se aclara que la utilización de este descargable se encuentra limitada de manera expresa para los propósitos educacionales del curso.

