
4

Técnicas de Relevamiento

Versión 1.0
Marzo 1999

Anabella Giuliano
Daniel Bustamante

4.1 Introducción

La ingeniería de requerimientos comprende cuatro procesos:

- *Obtención o relevamiento*: es el proceso a través del cual clientes o usuarios de un sistema descubren, articulan y entienden sus requerimientos.
- *Análisis*: es un proceso que consiste en examinar los requerimientos obtenidos en la etapa anterior, encontrar sus relaciones y descubrir inconsistencias, conflictos y requerimientos perdidos.
- *Especificación*: documentar requerimientos. Su representación puede ser de diversas formas: lenguaje natural, formal, simbólico o gráfico.
- *Validación*: confirmar con el usuario o cliente que los requerimientos especificados son correctos, válidos y completos.

4.1.1 Proceso de obtención de requerimientos:

El procedimiento de obtención de requerimientos consta en general de cinco pasos:

- Identificar la fuente de requerimientos, (usuarios).
- Realizar preguntas apropiadas para obtener un entendimiento de sus necesidades.
- Analizar la información obtenida, buscando inconsistencias o puntos aún no resueltos.
- Confirmar o validar lo entendido con los usuarios.
- Hacer una síntesis de los requerimientos.

4.1.1.1 Participantes:

- Ingeniero de software, quien es responsable de realizar la especificación de requerimientos.
- Los usuarios potenciales, quienes utilizarán el sistema directamente. También se encuentran involucrados los gerentes de departamento y ejecutivos de la compañía, especialmente los sponsors del proyecto.

4.1.1.2 Resultados de un buen proceso de obtención:

El resultado tangible es un conjunto de requerimientos que pueden ser usados por el equipo de desarrollo de software.

- Generalmente los usuarios que participan de este proceso tiene una vaga idea de lo que realmente necesitan y de lo que un sistema les puede ofrecer. Un buen proceso de obtención los ayuda a explorar y entender en profundidad sus requerimientos, y a saber diferenciar lo que quieren de lo que necesitan.
- Los usuarios tienen un buen conocimiento de la implicancia de las decisiones tomadas en la etapa de definición de requerimientos. Estos junto con el ingeniero de software comparten una visión de los problemas que ellos tratan de resolver y de las soluciones posibles. Se sienten satisfechos, bien informados y capacitados, y se comprometen con el éxito del proyecto.
- Los ingenieros y desarrolladores que participaron de este proceso resuelven el problema correcto para el usuario, tienen especificaciones claras del sistema a ser construido. Ellos obtienen un amplio conocimiento del dominio del sistema.

4.1.1.3 Resultados de un proceso pobre:

- El problema más serio es que los desarrolladores resuelven el problema equivocado. Esto termina en el fracaso total del proyecto.
- Muchas veces los desarrolladores no escuchan a los usuarios, o dominan el proceso y tienden a forzar sus propias interpretaciones y punto de vista en ellos. La consecuencia de esto es una participación menos efectiva de los usuarios en el proceso, le dedican menos tiempo y responden de forma incompleta a las preguntas del desarrollador.
- Muchas veces estos procesos terminan siendo caóticos: los desarrolladores descubren la pérdida de información valiosa y terminan en reuniones adicionales con los usuarios, pueden tomar decisiones equivocadas por malinterpretar las necesidades de ellos. A esto se suma que los requerimientos usualmente pueden cambiar y traer retrasos y esfuerzo invertido en diseño e implementación. El resultado de este caos es costos y tiempos mayores a los previstos.

4.1.1.4 Problemas que pueden presentarse durante este proceso:

4.1.1.4.1 Problemas de articulación:

El usuario puede no expresar correctamente sus necesidades y los desarrolladores malinterpretarlas.

- Los usuarios tienen problemas para expresar sus necesidades. Esto puede darse por no ser consciente de ellas, o bien conocerlas pero no saber expresarlas de manera apropiada.
- Los usuarios pueden ser conscientes de sus necesidades, pero tener miedo de expresarlas. El caso más común es con personas nuevas en la organización.
- Muchas veces los usuarios y desarrolladores entienden del mismo término diferente significado. Esto hace que muchos conceptos o relaciones se malinterpreten.
- Los usuarios no toman decisiones sobre algunos temas porque no entienden las alternativas o sus consecuencias.
- Una sola persona no conoce el problema completamente. Esto sucede especialmente en sistemas complejos.
- Los desarrolladores no escuchan a los usuarios y muchas veces pierden detalles importantes.
- Los desarrolladores no entienden el problema desde el punto de vista del usuario, no comprenden el contexto del mismo.
- Los desarrolladores tienden a anular o dominar a los usuarios.

4.1.1.4.2 Limitaciones de conocimiento:

- El analista no tiene un adecuado conocimiento del dominio; esto hace que tomen decisiones erróneas.
- Muchas veces se toman decisiones y luego se documentan mal, o la documentación se malinterpreta porque no recuerdan exactamente lo que se habló.
- Se utiliza información estadística para expresar requerimientos, que es interpretada de diferente manera por las distintas personas.
- Muchas personas tienden a simplificar el problema, y no siempre lo hacen de manera adecuada: pueden ignorar parte de los requerimientos.
- Algunos desarrolladores tienden a focalizarse en un punto que creen es el más importante y ellos entienden mejor.

4.1.1.4.3 Comportamiento humano:

- Conflictos y ambigüedades en los roles: cada usuario puede asumir que es responsabilidad de otro comentarle algún tema al desarrollador, y en definitiva nadie se lo dice. El desarrollador puede asumir que un usuario domina completamente el problema y el usuario puede asumir que el desarrollador le ha formulado todas las preguntas necesarias. Esto hace que queden muchas cosas por analizarse.
- Temor a cambios organizacionales y a la posible pérdida de trabajo. Muchas veces los usuarios no se comprometen con el proyecto por este miedo.

4.1.1.4.4 Aspectos técnicos:

- Problemas a ser resueltos de alta complejidad. Éstos están basados prácticamente en el conocimiento del usuario experto en el dominio.
- Los requerimientos generalmente cambian; hay que evitar que se vuelvan obsoletos cuando esta etapa está completa.
- Las tecnologías de hardware y software cambian rápidamente.
- Existen distintas fuentes de requerimientos; los usuarios del sistema no son conscientes de todo lo que el sistema debe satisfacer. Hay otros aspectos a tener en cuenta que no son requerimientos del usuario como estándares de industrias, regulaciones del gobierno, marketing, etc.
- Un nuevo sistema que es muy similar a otros construidos previamente tiene menor esfuerzo en la etapa de obtención de requerimientos que uno sin precedentes.

4.2 Técnica de Entrevistas

Es una técnica de obtención de requerimientos muy importante para extraer información detallada. En sistemas grandes se utiliza como una parte de las técnicas de alto nivel y en pequeños proyectos se puede utilizar como única técnica.

4.2.1 Enfoque clásico:

Primero el analista prepara un conjunto de preguntas para la entrevista. Luego se lleva a cabo la entrevista, generalmente entrevistando un usuario por vez y tomando notas luego de la reunión. Como último paso se revisan las notas y se crea el modelo. Este enfoque se utiliza mucho pero produce grandes demoras y pérdidas de tiempo importantes.

4.2.2 Enfoque estructurado:

El primer paso es construir modelos iniciales, utilizando DFD, DER, DTE, etc. El segundo paso es organizar y llevar a cabo las entrevistas; los modelos construidos se usan como base de la discusión, se utiliza el mismo modelo para varias entrevistas con lo cual es revisado por distintas personas. El tercer paso es la fase de análisis; se pueden contactar usuarios nuevamente por algunas dudas o información perdida. El cuarto paso es relacionar todos los modelos construidos, es el cierre de la fase.

4.2.3 Enfoque iterativo:

Reemplaza al enfoque secuencial. También se construyen modelos iniciales como base de las entrevistas. Luego de cada entrevista el modelo revisado es enviado al entrevistado para su aprobación, y éste modelo va a ser la base de la próxima entrevista. De esta manera los modelos son continuamente refinados y actualizados durante las entrevistas. Una de las principales ventajas es que la fase de análisis es mucho más corta pues parte del análisis se realiza en la etapa de las entrevistas, y además la revisión es mucho más ágil ya que el usuario cuenta con más información. El resultado es mejor productividad, eficiencia, y menor costo.

4.2.4 Fases de la técnica de entrevistas:

Se divide en cuatro fases: Identificar candidatos para entrevistar, Preparar la entrevista, Conducir la entrevista y Actividades de seguimiento (follow-up).

4.2.4.1 Identificar candidatos para entrevistar:

Generalmente se comienza con la persona que autorizó la construcción del sistema (sponsor), suelen ser gerentes o ejecutivos. El organigrama de la organización ayuda a identificar a otras personas relevantes como aquellos que reportan a los gerentes. Ellos tienen una amplia visión de por qué el sistema se está construyendo, y quiénes son los usuarios finales.

A cada entrevistado se le debe preguntar qué otras personas podrían o deberían ser entrevistadas.

También deben tenerse en cuenta personas que no van a utilizar directamente el sistema pero que interactúan con los usuarios del sistema.

4.2.4.2 Preparar la entrevista:

En esta fase el analista debe ponerse de acuerdo con el entrevistado en la agenda de la entrevista y preparar una lista de preguntas. Las entrevistas deben ser agendadas con suficiente tiempo, deben estar de acuerdo en la duración, y se le debe dar al entrevistado una guía de los temas que se van a tocar para que esté preparado. Muchas veces se utilizan grabadores, pero esto generalmente pone nervioso al entrevistado, entonces se le debe pedir permiso para usarlos.

Para preparar la lista de preguntas se utiliza una guía con ideas generales del sistema que se va a construir, y una vez redactadas se agrupan por tema, y se define cuanto tiempo se le va a dedicar a cada punto.

4.2.4.3 Conducir la entrevista:

Para comenzar la entrevista deben revisarse los objetivos de ésta, los temas que van a ser cubiertos, y el tiempo que se le va a dedicar a cada punto. Si se van a utilizar diagramas o modelos deben explicarse al comienzo.

Probablemente el analista deberá explorar en algunas respuestas un tanto ambiguas. Esto puede realizarse haciendo resumen, reformulando alguna pregunta, o explicar lo entendido para que el usuario lo confirme. Es importante hacer resúmenes durante el transcurso de la entrevista, no solo al final.

Se le debe dar al usuario la oportunidad de explicar en detalle sus repuestas, tratar de no hacer preguntas cuyas respuestas sean “sí” o “no”. Éstas son útiles para completar o confirmar alguna respuesta.

Evitar la tendencia a anticipar las respuestas.

Debe tratar de detectase la mayor cantidad de errores posibles, chequearlos con el usuario y corregirlos; esto garantiza la performance de la entrevista.

La entrevista puede terminar una vez que todas las preguntas están respondidas, el tiempo acordado fue superado, o cuando el desarrollador siente que ya no puede obtener más información relevante.

Al final siempre es bueno hacer un resumen de todo lo visto.

4.2.4.4 Actividades de follow up:

Una vez finalizada la entrevista debe realizarse un resumen escrito de toda información obtenida, y enviarla al usuario para que la revea también y la confirme.

Esto es muy útil para detectar ambigüedades, temas que necesitan analizarse mas profundamente e información perdida.

4.3 Técnica Brainstorming

4.3.1 Objetivo

Esta técnica es útil para plantear el problema desde diferentes puntos de vista, y obtener distintos caminos para resolverlo. Ayuda a construir una visión mas completa de las necesidades del usuario, y tiene la ventaja de que es fácil de aprender.

4.3.2 Características

En la sesión participan de cuatro a diez personas. Una persona es el líder.

Para preparar una sesión primero se identifican los participantes: clientes, compradores, los usuarios que van a utilizar el software y los ingenieros de software que lo van a desarrollar.

El resultado depende directamente de las ideas generadas por los participantes, por eso es necesario incluir personas con conocimiento y experiencia apropiada al sistema que va a ser construido.

Existen algunas herramientas que permiten llevar a cabo estas sesiones sin necesidad de que todas las personas estén en el mismo lugar físico, por ejemplo videoconferencia.

La sesión de brainstorming consiste de dos fases:

4.3.2.1 Fase de generación:

El líder abre la sesión realizando una breve explicación del problema a resolver y a partir de ahí cada participante expone sus ideas, sin que nadie opine ni discuta su validez.

La forma en que se va a llevar a cabo estas exposiciones queda a criterio del líder.

Hay que tener en cuenta las siguientes reglas:

- los participantes deben sentirse con total libertad para expresar sus ideas
- cuanto más cantidad de ideas se propongan, la calidad de las mismas va a ser mayor
- para complementar las nuevas ideas los participantes deben combinar o mejorar las ideas de los demás. Para poder lograr esto se propone que una persona tome nota de todas las ideas en un pizarrón o cartulina a la vista de todos, y también facilitar hojas a cada uno para que escriba sus propias ideas en el momento que le surjan.

La fase de generación puede concluir cuando el líder vea que no se están generando buenas ideas, en este caso se pospone la sesión; o bien cuando ya se hayan generado suficientes ideas, en este caso se pasa a la próxima fase.

4.3.2.2 Fase de consolidación:

Se evalúan las ideas generadas en la etapa anterior. El primer paso es revisar las ideas para clarificar; en algunos casos es necesario reformularlas o combinar algunas de ellas. A partir de aquí van analizando cada una y descartando las que no es posible llevarlas a la práctica dentro del contexto que se está analizando. Las restantes se priorizan, y se identifican las esenciales. Luego todo es documentado.