



# Software Requirements

## Third Edition

Best practices



Karl Wieggers and Joy Beatty

## 7. “Elicitación” de requisitos

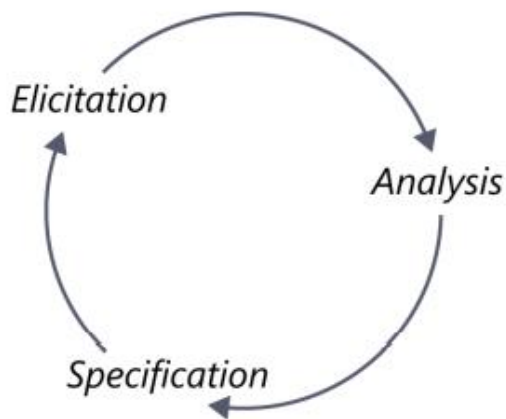
El corazón del desarrollo de requisitos es la “elicitación”, el proceso de identificación de las necesidades y limitaciones de los diversos interesados para un sistema de software. “elicitación” (obtención) no es lo mismo que “recopilación de requisitos”. Tampoco es una simple cuestión de transcribir exactamente lo que dicen los usuarios. La “elicitación” es un proceso colaborativo y analítico que incluye actividades para recopilar, descubrir, extraer y definir requisitos. La “elicitación” se utiliza para descubrir requisitos empresariales, de usuario, funcionales y no funcionales, junto con otros tipos de información. La obtención de requisitos es quizás el aspecto más desafiante, crítico, propenso a errores y de comunicación intensiva del desarrollo de software.

Involucrar a los usuarios en el proceso de “elicitación” es una forma de obtener apoyo y aceptación para el proyecto. Si usted es el analista de negocios, intente comprender los procesos de pensamiento detrás de los requisitos que establecen los usuarios. Recorra los procesos que siguen los usuarios para tomar decisiones sobre su trabajo y extraiga la lógica subyacente. Asegúrese de que todos entiendan por qué el sistema debe realizar ciertas funciones. Busque requisitos propuestos que reflejen reglas o procesos comerciales obsoletos o ineficaces que no deberían incorporarse a un nuevo sistema.

El BA debe crear un entorno propicio para una exploración exhaustiva del producto que se especifica. Para facilitar una comunicación clara, utilice el vocabulario del dominio empresarial en lugar de obligar a los clientes a comprender la jerga técnica. Registre los términos significativos del dominio de la aplicación en un glosario, en lugar de asumir que todos los participantes comparten las mismas definiciones. Los clientes deben comprender que una discusión sobre la posible funcionalidad no es un compromiso para incluirla en el producto. La lluvia de ideas y la imaginación de las posibilidades es un asunto aparte del análisis de las prioridades, la viabilidad y las realidades limitantes. Nunca es demasiado pronto para que las partes interesadas prioricen sus listas de deseos para evitar definir un proyecto enorme que nunca ofrece nada útil.

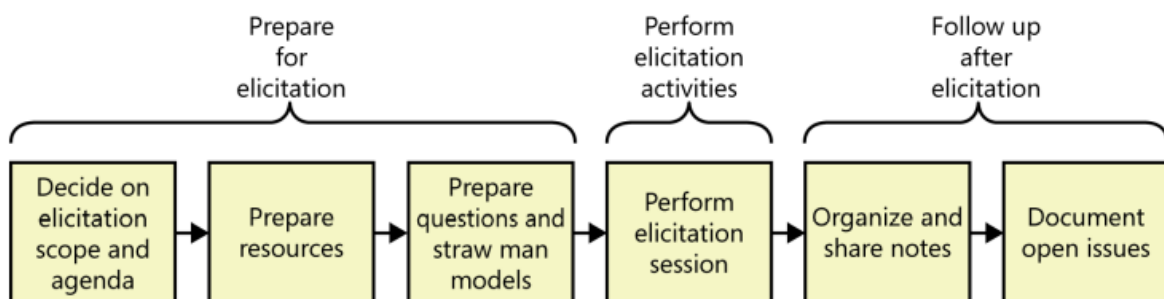
El resultado del desarrollo de requisitos es una comprensión común de las necesidades de los diversos interesados en el proyecto. Cuando los desarrolladores entienden esas necesidades, pueden explorar soluciones alternativas para abordarlas. Los participantes en la “elicitación” deben resistir la tentación de diseñar el sistema hasta que entiendan el problema. De lo contrario, pueden esperar hacer un trabajo de diseño considerable a medida que los requisitos se definan mejor. Enfatizar las tareas del usuario en lugar de las interfaces de usuario y centrarse en las verdaderas necesidades más que en los deseos expresados ayuda a evitar que el equipo se desvíe al especificar prematuramente los detalles del diseño.

Como muestra la figura 7-1, la naturaleza del desarrollo de requisitos es cíclica. Hará alguna “elicitación”, estudiará lo que aprendió, escribirá algunos requisitos, tal vez determine que le falta alguna información, realice una “elicitación” adicional, etc. No espere simplemente realizar un par de talleres de “elicitación” y luego declarar la victoria y seguir adelante.



**FIGURE 7-1** The cyclic nature of requirements elicitation, analysis, and specification.

Este capítulo describe una variedad de técnicas de “elicitación” efectivas, incluido cuándo usar cada una, así como consejos y desafíos para cada una. El resto del capítulo describe el proceso general de “elicitación”, desde la planificación de las actividades de “elicitación” hasta la organización de los resultados de la sesión. Más adelante en el capítulo, ofrecemos advertencias sobre algunas trampas a tener en cuenta durante la “elicitación” y sugerencias específicas para identificar los requisitos que faltan. La Figura 7-2 muestra las actividades para una única sesión de “elicitación” de requisitos. Sin embargo, antes de pasar por este proceso, exploremos algunas de las técnicas de obtención de requisitos que podría encontrar valiosas.



**FIGURE 7-2** Activities for a single requirements elicitation session.

## Técnicas de “elicitación” de requerimientos

Se pueden emplear numerosas técnicas de “elicitación” en proyectos de software. De hecho, ningún equipo de proyecto debería esperar usar una sola técnica de “elicitación”. Siempre hay muchos tipos de información por descubrir, y diferentes partes interesadas preferirán diferentes enfoques. Un usuario puede articular claramente cómo usa el sistema, mientras que es posible que deba observar a otro realizar su trabajo para alcanzar el mismo nivel de comprensión.

Las técnicas de “elicitación” incluyen ambas actividades facilitadas, en las que interactúa con las partes interesadas para obtener requisitos, y actividades independientes, en las que trabaja por su cuenta para descubrir información. Las actividades facilitadas se centran principalmente en descubrir los requisitos del negocio y del usuario. Es necesario trabajar directamente con los usuarios porque los requisitos de los usuarios abarcan las tareas que los usuarios deben realizar con el sistema. Para obtener los requisitos comerciales, deberá trabajar con personas como el patrocinador del proyecto. Las técnicas de “elicitación” independientes complementan los requisitos que presentan los usuarios y revelan la funcionalidad necesaria que los usuarios finales pueden no conocer.

La mayoría de los proyectos utilizarán una combinación de actividades de “elicitación” facilitadas e independientes. Cada técnica ofrece una exploración diferente de los requisitos o incluso puede revelar requisitos completamente diferentes. Las siguientes secciones describen varias técnicas comúnmente utilizadas para obtener requisitos.

- [Entrevistas](#)
- [Talleres de trabajo](#)
- [Grupos focales](#)
- [Análisis de interfaz del sistema](#)
- [Análisis de la interfaz de usuario](#)
- [Cuestionarios](#)
- [Análisis de documentos](#)
- [Observaciones](#)

## Entrevistas

La forma más obvia de averiguar qué necesitan los usuarios de un sistema de software es preguntarles. Las entrevistas son una fuente tradicional de entrada de requisitos tanto para productos comerciales como para sistemas de información, en todos los enfoques de desarrollo de software. La mayoría de los BA facilitará alguna forma de entrevistas individuales o en grupos pequeños para obtener los requisitos de sus proyectos. Los proyectos ágiles hacen un uso extensivo de las entrevistas como mecanismo para lograr la participación directa del usuario. Las entrevistas son más fáciles de programar y dirigir que las actividades de grupos grandes, como los talleres de requisitos.

Si es nuevo en un dominio de aplicación, las entrevistas con expertos pueden ayudarlo a ponerse al día rápidamente. Esto le permitirá preparar borradores de requisitos y modelos para usar en otras entrevistas o en talleres. Si puede establecer una buena relación con los entrevistados, se sentirán más seguros cuando compartan sus pensamientos uno a uno o en un grupo pequeño que en un taller más grande, particularmente sobre temas delicados. También es más fácil lograr que los usuarios acepten participar en el proyecto o revisar los requisitos existentes durante una entrevista individual o en un grupo pequeño que en un entorno de grupo grande. Las entrevistas son apropiadas para obtener requisitos comerciales de ejecutivos que no tienen mucho tiempo para reunirse. Para obtener orientación sobre cómo realizar entrevistas con usuarios, consulte Ian Alexander y Ljerka Beus-Dukic (2009) y Howard Podeswa (2009). A continuación se presentan algunas sugerencias para realizar entrevistas. Estos son también consejos útiles para realizar talleres de “elicitación”.

### **Establecer una buena relación.**

Para comenzar una entrevista, preséntese si los asistentes aún no lo conocen, revise la agenda, recuerde a los asistentes los objetivos de la sesión y aborde cualquier pregunta preliminar o inquietud que tengan los asistentes.

### **Manténgase en el alcance.**

Al igual que con cualquier sesión de “elicitación”, mantenga la discusión enfocada en su objetivo. Incluso cuando está hablando con una sola persona o un grupo pequeño, existe la posibilidad de que la entrevista se salga del tema.

### **Prepare preguntas y modelos de hombre de paja con anticipación.**

Prepárese para las entrevistas redactando cualquier material que pueda de antemano, como una lista de preguntas para guiar la conversación. Los borradores de materiales brindarán a sus usuarios un punto de partida para pensar. Las personas a menudo pueden criticar el contenido más fácilmente de lo que pueden crearlo. La preparación de preguntas y la elaboración de modelos de testafierro se describen con más detalle en la sección “Preparación para la “elicitación”” más adelante en este capítulo.

## Sugerir ideas

En lugar de simplemente transcribir lo que dicen los clientes, un BA creativo propone ideas y alternativas durante la “elicitación”. A veces, los usuarios no se dan cuenta de las capacidades que pueden proporcionar los desarrolladores; es posible que se entusiasmen cuando sugiera una funcionalidad que hará que el sistema sea especialmente valioso. Cuando los usuarios realmente no pueden expresar lo que necesitan, quizás pueda verlos trabajar y sugerir formas de automatizar partes del trabajo (consulte la sección “Observaciones” más adelante en este capítulo). Los BA pueden pensar fuera de la caja mental que limita a las personas que están demasiado cerca del dominio del problema.

## Escucha activamente

Practique las técnicas de escucha activa (inclinarse hacia adelante, mostrar paciencia, dar retroalimentación verbal y preguntar cuando algo no está claro) y parafrasear (reiterar la idea principal del mensaje de un orador para mostrar su comprensión de ese mensaje).

## Talleres de trabajo

Los talleres fomentan la colaboración de las partes interesadas en la definición de requisitos. Ellen Gottesdiener (2002) define un taller de requisitos como “una reunión estructurada en la que un grupo cuidadosamente seleccionado de partes interesadas y expertos en contenido trabajan juntos para definir, crear, refinar y llegar al cierre de los entregables (como modelos y documentos) que representan los requisitos del usuario. .” Los talleres son sesiones facilitadas con múltiples partes interesadas y roles formales, como un facilitador y un escriba. Los talleres a menudo incluyen varios tipos de partes interesadas, desde usuarios hasta desarrolladores y evaluadores. Se utilizan para obtener requisitos de múltiples partes interesadas al mismo tiempo. Trabajar en grupo es más efectivo para resolver desacuerdos que hablar con las personas individualmente. Además, los talleres son útiles cuando se necesita una respuesta rápida a la “elicitación” debido a limitaciones de horario.

Según una autoridad, “La facilitación es el arte de guiar a las personas a través de procesos hacia objetivos acordados de una manera que fomente la participación, la propiedad y la productividad de todos los involucrados” (Sibbet 1994). El facilitador juega un papel fundamental en la planificación del taller, la selección de los participantes y su orientación hacia un resultado exitoso. Los analistas de negocios frecuentemente facilitan talleres de “elicitación”. Cuando un equipo está comenzando con nuevos enfoques para la obtención de requisitos, considere tener un facilitador externo o un segundo BA que facilite los talleres iniciales. De esta manera, el BA principal puede dedicar toda su atención a la discusión. Si el BA único también actúa como facilitador, debe tener en cuenta cuándo habla como facilitador y cuándo participa en la discusión. Un escriba ayuda al facilitador capturando los puntos que surgen durante la discusión. Es extremadamente desafiante facilitar, redactar y participar simultáneamente y hacer un buen trabajo en los tres.

Los talleres pueden requerir muchos recursos, a veces requieren numerosos participantes durante varios días a la vez. Deben estar bien planificados para no perder el tiempo. Minimice la pérdida de tiempo viniendo a un taller con borradores de materiales preparados con anticipación. Por ejemplo, puede redactar casos de uso que se pueden revisar como grupo en lugar de que todo el grupo los redacte juntos. Rara vez tiene sentido comenzar un taller con una pizarra completamente en blanco. Utilice otras técnicas de “elicitación” antes de los talleres y luego reúna a las partes interesadas para trabajar solo en las áreas necesarias.

Las prácticas generales de facilitación se aplican a la obtención de requisitos (Schwarz 2002). Un recurso definitivo específico para facilitar talleres de “elicitación” de requisitos es Requisitos de colaboración de Gottesdiener (2002). Ella describe una gran cantidad de técnicas y herramientas para la facilitación de talleres. Los siguientes son algunos consejos para realizar talleres de “elicitación” efectivos, muchos de los cuales también se aplican a las entrevistas.

### **Establecer y hacer cumplir las reglas básicas**

Los participantes del taller deben ponerse de acuerdo sobre algunos principios operativos básicos. Los ejemplos incluyen comenzar y terminar a tiempo; regresar de los descansos puntualmente; silenciar dispositivos electrónicos; mantener una conversación a la vez; esperar que todos contribuyan; y centrar los comentarios y las críticas en temas más que en individuos. Una vez establecidas las reglas, asegúrese de que los participantes las sigan.

### **Cumple todos los roles del equipo.**

Un facilitador debe asegurarse de que las siguientes tareas estén cubiertas por personas en el taller: tomar notas, controlar el tiempo, administrar el alcance, administrar las reglas básicas y asegurarse de que todos sean escuchados. Un escriba puede registrar lo que está pasando, mientras que otra persona mira el reloj

### **Planifica una agenda.**

Cada taller necesita un plan claro, como se explica en la sección "Preparación para la “elicitación”" más adelante en este capítulo. Cree el plan y la agenda del taller con anticipación y comuníquese a los participantes para que sepan los objetivos y qué esperar y puedan prepararse en consecuencia.

### **Manténgase en el alcance (Stay in scope)**

Consulte los requisitos comerciales para confirmar si los requisitos del usuario propuestos se encuentran dentro del alcance del proyecto actual. Mantenga cada taller enfocado en el nivel correcto de abstracción para los objetivos de esa sesión. Los grupos se sumergen fácilmente en detalles que distraen durante las discusiones de requisitos. Esas discusiones consumen tiempo que el grupo debería dedicar a desarrollar una comprensión de mayor

nivel de los requisitos del usuario; los detalles vendrán más tarde. El facilitador tendrá que atraer a los participantes de “elicitación” periódicamente para mantenerlos en el tema.

### **Trampa**

Tenga cuidado con las discusiones fuera del tema, como las exploraciones de diseño, durante las sesiones de “elicitación”. Mantenga a los participantes enfocados en los objetivos de la sesión, mientras les asegura que tendrán oportunidades futuras para resolver otros problemas que surjan.

### **Use estacionamientos para capturar artículos para su consideración posterior**

Una variedad de información aleatoria pero importante surgirá durante las discusiones de “elicitación”: atributos de calidad, reglas comerciales, ideas de interfaz de usuario y más. Organice esta información en rotafolios —estacionamientos— para no perderla y demostrar respeto por el participante que la mencionó. No se distraiga discutiendo detalles fuera del tema a menos que resulten ser sensacionales. Describa lo que sucederá con los problemas del estacionamiento después de la reunión.

### **Debates de la caja de tiempo**

Considere asignar un período fijo de tiempo a cada tema de discusión. Es posible que la discusión deba completarse más tarde, pero el timeboxing ayuda a evitar la trampa de dedicar mucho más tiempo del previsto al primer tema y descuidar por completo otros temas importantes. Al cerrar una discusión con límite de tiempo, resuma el estado y los próximos pasos antes de abandonar el tema.

### **Mantenga el equipo pequeño pero incluya a las partes interesadas adecuadas.**

Los grupos pequeños pueden trabajar mucho más rápido que los equipos más grandes. Los talleres de “elicitación” con más de cinco o seis participantes activos pueden verse envueltos en viajes secundarios, conversaciones simultáneas y disputas. Considere la posibilidad de realizar varios talleres en paralelo para explorar los requisitos de las diferentes clases de usuarios. Los participantes del taller podrían incluir al campeón del producto y otros representantes de los usuarios, tal vez un experto en la materia, un BA, un desarrollador y un evaluador. El conocimiento, la experiencia y la autoridad para tomar decisiones son calificaciones para participar en talleres de “elicitación”.

### **Demasiados cocineros**

Los talleres de “elicitación” de requisitos que involucren a demasiados participantes pueden ralentizarse hasta convertirse en un rastreo polémico. Mi colega Debbie estaba frustrada por el lento progreso del primer taller de casos de uso que facilitó para un proyecto de sitio web. Los 12 participantes sostuvieron discusiones extensas sobre detalles innecesarios y no pudieron ponerse de acuerdo sobre cómo debería funcionar cada caso de uso. El progreso del equipo se aceleró muy bien cuando



Debbie redujo el número de participantes a unos seis que representaban los roles clave de analista, cliente, arquitecto de sistemas, desarrollador y diseñador visual. El taller perdió algo de información al usar un equipo más pequeño, pero la tasa de progreso compensó con creces esa pérdida. Los participantes del taller deben intercambiar información fuera de línea con colegas que no asisten y luego traer la información recopilada a los talleres.

### **Mantenga a todos comprometidos**

A veces, ciertos participantes dejarán de contribuir a la discusión. Estas personas pueden sentirse frustradas por una variedad de razones. Tal vez su aporte no se toma en serio porque otros participantes no encuentran interesantes sus inquietudes, o tal vez no quieren interrumpir el trabajo que el grupo ha completado hasta ahora. Tal vez la parte interesada que se ha retirado está cediendo a participantes más agresivos o a un analista dominante. El facilitador debe leer el lenguaje corporal (falta de contacto visual, inquietud, suspiros, mirar el reloj), entender por qué alguien se ha desconectado del proceso y tratar de volver a involucrar a la persona. Las señales visuales están ausentes cuando facilita a través de una teleconferencia, por lo que debe escuchar con atención para saber quién no está participando y los tonos que se utilizan. Puede preguntar directamente a estas personas silenciosas si tienen alguna idea sobre la discusión que les gustaría compartir. El facilitador debe asegurarse de que todos sean escuchados.

### **Grupos focales**

Un grupo de enfoque es un grupo representativo de usuarios que se reúnen en una actividad de “elicitación” facilitada para generar aportes e ideas sobre los requisitos funcionales y de calidad de un producto. Las sesiones de grupos focales deben ser interactivas, permitiendo que todos los usuarios tengan la oportunidad de expresar sus pensamientos. Los grupos focales son útiles para explorar las actitudes, impresiones, preferencias y necesidades de los usuarios (IIBA 2009). Son particularmente valiosos si está desarrollando productos comerciales y no tiene fácil acceso a los usuarios finales dentro de su empresa.

### **Cuando estallan los conflictos**

Las diferentes perspectivas, prioridades y personalidades pueden generar conflictos e incluso enojo dentro de un grupo. Si esto sucede, trátelo de inmediato. Busque pistas no verbales que muestren conflicto o enojo y trate de entender la causa. Cuando el grupo tenga claro el motivo del conflicto, es posible que pueda encontrar una solución (si es necesario).

Si un individuo simplemente no participará de manera productiva, hable con él en privado para determinar si su presencia impedirá que el grupo avance. Si es así, es posible que deba agradecer a la persona por su tiempo y continuar sin él. A veces,

esta no será una opción y simplemente deberá abandonar la sesión o el tema por completo por ahora. La gestión de conflictos es una habilidad compleja de desarrollar y existen numerosos recursos al respecto (Fisher, Ury y Patton 2011; Patterson et al. 2011).

Una vez programé una sesión para obtener los requisitos comerciales de un nuevo director de ventas. Se sabía que tenía una personalidad antagónica, así que llegué a la reunión preparado para escuchar y comprender realmente sus deseos. En el primer minuto de la reunión, comenzó a gritarme, preguntándome por qué celebrábamos esta reunión. Él dijo: "¿Quién es usted para pensar que tiene derecho a preguntarme sobre mis objetivos comerciales?" Tomé una respiración profunda y una larga pausa. Entonces traté de explicarle por qué necesitaba entender sus objetivos comerciales: que sin ellos, el equipo estaría adivinando lo que necesitábamos desarrollar para satisfacer los deseos de los clientes, y él estaría muy decepcionado con los resultados. Y tan rápido como se enojó, lo superó. Sin dudarlo, comenzó a recitar sus objetivos comerciales. Afortunadamente, mi escriba estaba allí para atraparlos porque todavía estaba un poco desconcertado por todo el intercambio.

A menudo, tendrá una base de usuarios grande y diversa para aprovechar, así que seleccione los miembros del grupo de enfoque con cuidado. Incluya usuarios que hayan usado versiones anteriores o productos similares al que está implementando. Seleccione un grupo de usuarios que sean del mismo tipo (y mantenga varios grupos de enfoque para las diferentes clases de usuarios) o seleccione un grupo que represente el espectro completo de clases de usuarios para que todos estén igualmente representados.

Los grupos focales deben ser facilitados. Deberá mantenerlos en el tema, pero sin influir en las opiniones que se expresan. Es posible que desee grabar la sesión para poder volver atrás y escuchar atentamente los comentarios. No espere un análisis cuantitativo de los grupos focales, sino una gran cantidad de comentarios subjetivos que se pueden evaluar y priorizar más a medida que se desarrollan los requisitos. Las sesiones de "elicitación" con grupos focales se benefician de muchos de los mismos consejos descritos anteriormente para los talleres. Los participantes en los grupos focales normalmente no tienen autoridad para tomar decisiones sobre los requisitos.

## Análisis de la interfaz del sistema

El análisis de interfaz es una técnica de "elicitación" independiente que implica examinar los sistemas a los que se conecta su sistema. El análisis de la interfaz del sistema revela requisitos funcionales relacionados con el intercambio de datos y servicios entre sistemas (IIBA 2009). Los diagramas de contexto y los mapas de ecosistemas (consulte el Capítulo 5, "Establecimiento de los requisitos comerciales") son una opción obvia para comenzar a encontrar interfaces para estudios posteriores. De hecho, si encuentra una interfaz que

tiene requisitos asociados y que no está representada en uno de estos diagramas, los diagramas están incompletos.

Para cada sistema que interactúe con el suyo, identifique la funcionalidad en el otro sistema que podría generar requisitos para su sistema. Estos requisitos podrían describir qué datos pasar al otro sistema, qué datos se reciben de él y las reglas sobre esos datos, como los criterios de validación. También puede descubrir la funcionalidad existente que no necesita implementar en su sistema. Suponga que pensó que necesitaba implementar reglas de validación para un pedido de carrito de compras en un sitio web de comercio electrónico antes de pasarlo a un sistema de gestión de pedidos. A través del análisis de la interfaz del sistema, puede aprender que varios sistemas pasan pedidos al sistema de gestión de pedidos, que realiza la validación, por lo que no necesita crear esta función.

## Análisis de la interfaz de usuario

El análisis de la interfaz de usuario (UI) es una técnica de “elicitación” independiente en la que se estudian los sistemas existentes para descubrir los requisitos funcionales y del usuario. Es mejor interactuar directamente con los sistemas existentes, pero si es necesario, puede usar capturas de pantalla. Los manuales de usuario para implementaciones de software empaquetado comprado a menudo contienen capturas de pantalla que funcionarán bien como punto de partida. Si no existe un sistema, es posible que pueda ver las interfaces de usuario de productos similares.

Cuando se trabaja con soluciones empaquetadas o un sistema existente, el análisis de la interfaz de usuario puede ayudarlo a identificar una lista completa de pantallas para ayudarlo a descubrir funciones potenciales. Al navegar por la interfaz de usuario existente, puede conocer los pasos comunes que los usuarios toman en el sistema y redactar casos de uso para revisar con los usuarios. El análisis de la interfaz de usuario puede revelar datos que los usuarios necesitan ver. Es una excelente manera de ponerse al día sobre cómo funciona un sistema existente (a menos que necesite mucha capacitación para hacerlo). En lugar de preguntar a los usuarios cómo interactúan con el sistema y qué pasos toman, tal vez usted mismo pueda llegar a un entendimiento inicial.

No asuma que se necesita cierta funcionalidad en el nuevo sistema solo porque la encontró en uno existente. Además, no asuma que debido a que la interfaz de usuario se ve o fluye de cierta manera en el sistema actual, debe implementarse de esa manera en el sistema futuro.

## Cuestionarios

Los cuestionarios son una forma de encuestar a grandes grupos de usuarios para comprender sus necesidades. Son económicos, lo que los convierte en una opción lógica para obtener información de grandes poblaciones de usuarios, y se pueden administrar fácilmente a través de fronteras geográficas. Los resultados analizados de los cuestionarios

se pueden utilizar como entrada para otras técnicas de “elicitación”. Por ejemplo, puede usar un cuestionario para identificar los puntos débiles más grandes de los usuarios con un sistema existente y luego usar los resultados para discutir la priorización con los tomadores de decisiones en un taller. También puede usar cuestionarios para encuestar a los usuarios de productos comerciales para obtener comentarios.

Preparar preguntas bien escritas es el mayor desafío con los cuestionarios. Hay muchos consejos disponibles para escribir cuestionarios (Universidad Estatal de Colorado 2013), y aquí sugerimos los más importantes:

- Proporcione opciones de respuesta que cubran el conjunto completo de respuestas posibles.
- Haga que las opciones de respuesta sean mutuamente excluyentes (sin superposiciones en los rangos numéricos) y exhaustivas (enumere todas las opciones posibles y/o tenga un espacio para escribir para una opción en la que no pensó).
- No formule una pregunta de manera que implique una respuesta “correcta”.
- Si usa escalas, utilícelas consistentemente a lo largo del cuestionario.
- Utilice preguntas cerradas con dos o más opciones específicas si desea utilizar los resultados del cuestionario para el análisis estadístico. Las preguntas abiertas permiten a los usuarios responder como quieran, por lo que es difícil buscar puntos en común en los resultados.
- Considere consultar con un experto en diseño y administración de cuestionarios para asegurarse de hacer las preguntas correctas a las personas adecuadas.
- Siempre pruebe un cuestionario antes de distribuirlo. Es frustrante descubrir demasiado tarde que una pregunta se formuló de manera ambigua o darse cuenta de que se omitió una pregunta importante.
- No haga demasiadas preguntas o la gente no responderá.

## Análisis de documentos

El análisis de documentos implica examinar cualquier documentación existente en busca de posibles requisitos de software. La documentación más útil incluye especificaciones de requisitos, procesos comerciales, colecciones de lecciones aprendidas y manuales de usuario para aplicaciones existentes o similares. Los documentos pueden describir los estándares corporativos o de la industria que deben seguirse o las reglamentaciones que debe cumplir el producto. Al reemplazar un sistema existente, la documentación anterior puede revelar la funcionalidad que podría ser necesario conservar, así como la funcionalidad obsoleta. Para las implementaciones de soluciones empaquetadas, la documentación del proveedor menciona la funcionalidad que sus usuarios podrían

necesitar, pero es posible que deba explorar más a fondo cómo implementarla en el entorno de destino. Las revisiones comparativas señalan las deficiencias de otros productos que podría abordar para obtener una ventaja competitiva. Los informes de problemas y las solicitudes de mejora recopilados de los usuarios por el servicio de asistencia técnica y el personal de soporte de campo pueden ofrecer ideas para mejorar el sistema en versiones futuras.

El análisis de documentos es una forma de ponerse al día en un sistema existente o en un nuevo dominio. Investigar un poco y redactar algunos requisitos de antemano reduce el tiempo necesario para la reunión de “elicitación”. El análisis de documentos puede revelar información que la gente no le dice, ya sea porque no piensan en ella o porque no la conocen. Por ejemplo, si está creando una nueva aplicación de centro de llamadas, es posible que encuentre una lógica comercial complicada descrita en el manual del usuario para una aplicación existente. Quizás los usuarios ni siquiera conocen esta lógica. Puede utilizar los resultados de este análisis como entrada para las entrevistas con los usuarios.

Un riesgo de esta técnica es que los documentos disponibles pueden no estar actualizados. Los requisitos pueden haber cambiado sin que se actualicen las especificaciones, o se puede documentar una funcionalidad que no es necesaria en un nuevo sistema.

## Observaciones

Cuando les pide a los usuarios que describan cómo hacen su trabajo, es probable que les resulte difícil ser precisos: es posible que falten detalles o que sean incorrectos. A menudo, esto se debe a que las tareas son complejas y es difícil recordar cada detalle. En otros casos, se debe a que los usuarios están tan familiarizados con la ejecución de una tarea que no pueden articular todo lo que hacen. Quizás la tarea es tan habitual que ni siquiera piensan en ella. A veces se puede aprender mucho observando exactamente cómo los usuarios realizan sus tareas.

Las observaciones consumen mucho tiempo, por lo que no son adecuadas para todos los usuarios ni para todas las tareas. Para evitar interrumpir las actividades laborales asignadas regularmente a los usuarios, limite cada tiempo de observación a dos horas o menos. Seleccione tareas importantes o de alto riesgo y varias clases de usuarios para las observaciones. Si usa observaciones en proyectos ágiles, haga que el usuario demuestre solo las tareas específicas relacionadas con la próxima iteración.

Observar el flujo de trabajo de un usuario en el entorno de tareas le permite al BA validar la información recopilada de otras fuentes, identificar nuevos temas para entrevistas, ver problemas con el sistema actual e identificar formas en que el nuevo sistema puede respaldar mejor el flujo de trabajo. El BA debe abstraer y generalizar más allá de las actividades del usuario observado para garantizar que los requisitos capturados se apliquen a la clase de usuario en su conjunto, no solo a ese individuo.

Un BA hábil también puede sugerir ideas para mejorar los procesos comerciales actuales del usuario.

### **Mírame hornear un pastel**

Para demostrar el poder de las observaciones, cuéntales a algunos amigos los pasos para hornear un pastel a partir de una mezcla. Probablemente recordará los pasos para encender el horno, sacar los platos y utensilios necesarios, agregar cada ingrediente, mezclar los ingredientes, preparar la sartén, poner la masa en la sartén, hornearla y sacarla del horno. cuando termine. Pero cuando les dijiste a tus amigos que agregaran cada ingrediente, ¿te acordaste de decir que abrieran la bolsa que contenía la mezcla? ¿Recordaste decir romper la cáscara del huevo, agregar solo el contenido del huevo y desechar la cáscara? Estos pasos aparentemente obvios pueden no ser tan obvios para alguien que nunca antes ha horneado.

Las observaciones pueden ser silenciosas o interactivas. Las observaciones silenciosas son apropiadas cuando no se puede interrumpir a los usuarios ocupados. Las observaciones interactivas permiten que el BA interrumpa al usuario en medio de una tarea y le haga una pregunta. Esto es útil para comprender inmediatamente por qué un usuario hizo una elección o para preguntarle en qué estaba pensando cuando realizó alguna acción. Documente lo que observa para su posterior análisis después de la sesión. También puede considerar grabar la sesión en video, si las políticas lo permiten, para que pueda actualizar su memoria más tarde.

Estaba desarrollando una aplicación de centro de atención telefónica para representantes de servicio al cliente (CSR) que estaban acostumbrados a tener que hojear catálogos impresos para encontrar los productos que los clientes querían pedir. El equipo de BA se reunió con varios CSR para obtener casos de uso para la nueva aplicación. Cada uno dijo lo difícil que era tener que hojear múltiples catálogos para encontrar exactamente a qué producto se refería un cliente. Cada BA se sentó con un CSR diferente mientras los CSR tomaban pedidos por teléfono. Vimos la dificultad a la que se enfrentaban al verlos primero tratar de encontrar el catálogo por fecha y luego tratar de localizar el producto correcto. Las sesiones de observación nos ayudaron a comprender qué funciones necesitarían en un catálogo de productos en línea.

## Planificación de la “elicitación” de su proyecto

Al principio de un proyecto, el analista de negocios debe planificar el enfoque del proyecto para la obtención de requisitos. Incluso un simple plan de acción aumenta las posibilidades de éxito y establece expectativas realistas para las partes interesadas. Solo al obtener un compromiso explícito sobre los recursos de obtención, el cronograma y los entregables, puede evitar que los participantes se aparten para hacer otro trabajo. **Un plan de “elicitación” incluye las técnicas que usará, cuándo planea usarlas y con qué propósito.** Al igual que con cualquier plan, utilícelo como guía y recordatorio a lo largo del proyecto, pero tenga en cuenta que es posible que deba cambiar el plan a lo largo del proyecto. **Su plan debe abordar los siguientes elementos:**

■ **Objetivos de “elicitación”** Planifique los objetivos de “elicitación” para todo el proyecto y los objetivos para cada actividad de “elicitación” planificada.

■ **Estrategia de “elicitación” y técnicas planificadas** Decidir qué técnicas utilizar con los diferentes grupos de partes interesadas. Puede usar alguna combinación de cuestionarios, talleres, visitas a clientes, entrevistas individuales y otras técnicas, según el acceso que tenga a las partes interesadas, las limitaciones de tiempo y su conocimiento del sistema existente.

■ **Estimaciones de cronograma y recursos** Identifique tanto al cliente como a los participantes del desarrollo para las diversas actividades de obtención, junto con las estimaciones del esfuerzo y el tiempo de calendario requerido. Es posible que solo pueda identificar las clases de usuarios y no individuos específicos por adelantado, pero eso permitirá a los gerentes comenzar a planificar las próximas necesidades de recursos. Calcule el tiempo de BA, incluido el tiempo para prepararse para la “elicitación” y realizar el análisis de seguimiento.

■ **Documentos y sistemas necesarios para la obtención independiente** Si está realizando un análisis de documentos, interfaz de sistema o interfaz de usuario, identifique los materiales necesarios para asegurarse de tenerlos cuando los necesite.

■ **Productos esperados de los esfuerzos de “elicitación”** Saber que va a crear una lista de casos de uso, un SRS, un análisis de los resultados del cuestionario o especificaciones de atributos de calidad ayuda a garantizar que se dirija a las partes interesadas, los temas y los detalles correctos durante la “elicitación”.

■ **Riesgos de obtención** Identifique los factores que podrían impedir su capacidad para completar las actividades de obtención según lo previsto, calcule la gravedad de cada riesgo y decida cómo puede mitigarlo o controlarlo. Consulte el Capítulo 32, "Requisitos de software y gestión de riesgos", para obtener más información sobre la gestión de riesgos. Consulte el Apéndice B, "Guía de solución de problemas de requisitos", para conocer los

síntomas, las causas principales y las posibles soluciones para problemas comunes de obtención.

Muchos BA tienen su técnica de “elicitación” de “ir a”—comúnmente entrevistas y talleres—y no piensan usar otras técnicas que podrían reducir las necesidades de recursos o aumentar la calidad de la información descubierta. Rara vez un BA obtendrá los mejores resultados usando solo una técnica de “elicitación” en un proyecto. Las técnicas de “elicitación” se aplican en todo el espectro de enfoques de desarrollo. La selección de técnicas de “elicitación” debe basarse en las características del proyecto.

**La Figura 7-3 sugiere las técnicas de “elicitación” que probablemente sean útiles para varios tipos de proyectos.** Seleccione la fila o filas que representen las características de su proyecto y lea a la derecha para ver qué técnicas de “elicitación” tienen más probabilidades de ser útiles (marcadas con una X). Por ejemplo, si está desarrollando una nueva aplicación, es probable que obtenga los mejores resultados con una combinación de entrevistas con las partes interesadas, talleres y análisis de la interfaz del sistema. La mayoría de los proyectos pueden hacer uso de entrevistas y talleres. Los grupos focales son más apropiados que los talleres para software de mercado masivo porque tiene una gran base de usuarios externos pero acceso limitado a los representantes. Estas sugerencias de técnicas de “elicitación” son solo eso: sugerencias. Por ejemplo, puede concluir que desea aplicar el análisis de la interfaz de usuario en proyectos de software de mercado masivo.

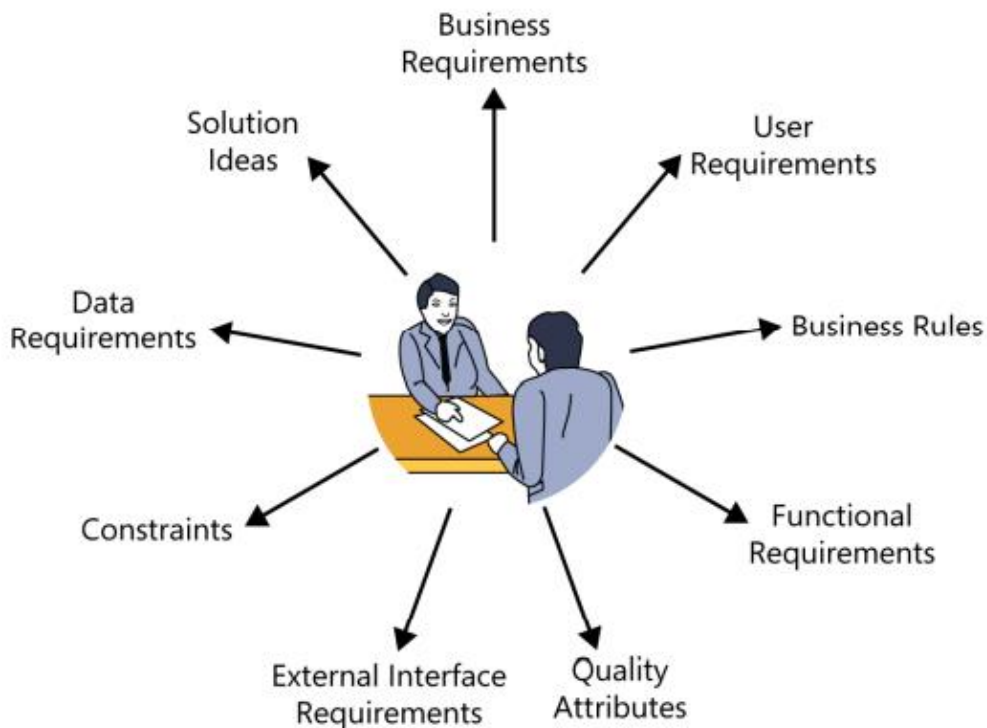
	Interviews	Workshops	Focus groups	Observations	Questionnaires	System interface analysis	User interface analysis	Document analysis
Mass-market software	x		x		x			
Internal corporate software	x	x	x	x		x		x
Replacing existing system	x	x		x		x	x	x
Enhancing existing system	x	x				x	x	x
New application	x	x				x		
Packaged software implementation	x	x		x		x		x
Embedded systems	x	x				x		x
Geographically distributed stakeholders	x	x			x			

**FIGURE 7-3** Suggested elicitation techniques by project characteristic.



## Clasificación de la entrada del cliente

No espere que sus clientes presenten una lista sucinta, completa y bien organizada de sus necesidades. Los analistas deben clasificar la miríada de información sobre requisitos que escuchan en varias categorías para poder documentarla y utilizarla adecuadamente. **La Figura 7-7 ilustra nueve de estas categorías.** Durante las actividades de “elicitación”, haga anotaciones rápidas en sus notas si detecta que alguna parte de la información es de este tipo. Por ejemplo, escriba "DD" en un pequeño círculo si reconoce una definición de datos



**FIGURE 7-7** Classifying customer input.

Al igual que con muchas categorizaciones, la información recopilada podría no encajar con precisión en estos nueve cubos. Probablemente le sobrará información después de esta clasificación. Cualquier cosa que no encaje en una de estas categorías podría ser:

- Un requisito del proyecto no relacionado con el desarrollo de software, como la necesidad de capacitar a los usuarios en el nuevo sistema.
- Una restricción del proyecto, como una restricción de costo o cronograma (a diferencia de las restricciones de diseño o implementación descritas en este capítulo).
- Una suposición o una dependencia.

- Información adicional de carácter histórico, contextual o descriptivo.
- Información superflua que no agrega valor.

Los participantes de la “elicitación” no le dirán simplemente: "Aquí viene un requisito de negocio". Como analista, debe determinar qué tipo de información representa cada declaración proporcionada que escucha. **La siguiente discusión sugiere algunas frases para escuchar que lo ayudarán en este proceso de clasificación.**

**Requisitos de negocio** Cualquier cosa que describa el beneficio financiero, de mercado u otro beneficio comercial que los clientes o la organización en desarrollo desean obtener del producto es un requisito de negocio (consulte el Capítulo 5). Preste atención a declaraciones sobre el valor que recibirán los compradores o usuarios del software, como estas:

- “Aumentar la participación de mercado en la región X en Y por ciento dentro de Z meses”.
- “Ahorre \$X por año en electricidad que ahora se desperdicia en unidades ineficientes”.

**Requisitos del usuario** Las declaraciones generales de los objetivos del usuario o las tareas de negocio que los usuarios deben realizar son requisitos del usuario, representados normalmente como casos de uso, escenarios o historias de usuarios (consulte el Capítulo 8, "Comprensión de los requisitos del usuario"). Un usuario que dice "Necesito <hacer algo>" probablemente esté describiendo un requisito del usuario, como en los siguientes ejemplos:

- “Necesito imprimir una etiqueta de envío para un paquete”.
- “Como operador principal de la máquina, necesito calibrar el controlador de la bomba a primera hora de cada mañana.”

**Reglas de negocio** Cuando un cliente dice que solo ciertos usuarios pueden realizar una actividad bajo condiciones específicas, podría estar presentando una regla de negocio (consulte el Capítulo 9, “Siguiendo las reglas”). Estos no son requisitos de software tal como están, pero puede derivar algunos requisitos funcionales para hacer cumplir las reglas. Frases como “Debe cumplir con . . .”, “Si <alguna condición es verdadera>, entonces <algo sucede>”, o “Debe calcularse de acuerdo con . . .” sugiere que el usuario está describiendo una regla de negocio. Aquí hay unos ejemplos:

- “Un nuevo cliente debe pagar el 30 por ciento de la tarifa de consultoría estimada y los gastos de viaje por adelantado”.
- “Las aprobaciones de tiempo libre deben cumplir con la política de vacaciones de recursos humanos de la empresa.

**Requisitos funcionales** Los requisitos funcionales describen los comportamientos observables que exhibirá el sistema bajo ciertas condiciones y las acciones que el sistema

permitirá que tomen los usuarios. Estos son algunos ejemplos de requisitos funcionales que puede escuchar de los usuarios:

- “Si la presión supera los 40,0 psi, la luz de advertencia de alta presión debería encenderse”.
- “El usuario debe poder ordenar la lista de proyectos en orden alfabético hacia adelante y hacia atrás”.

Estas declaraciones ilustran cómo los usuarios suelen presentar requisitos funcionales, pero no representan buenas formas de escribir requisitos funcionales. El BA deberá elaborar estos en especificaciones más precisas. Consulte el Capítulo 11, "Redacción de requisitos excelentes", para obtener orientación sobre la redacción de buenos requisitos funcionales.

**Atributos de calidad** Las declaraciones que describen qué tan bien el sistema hace algo son atributos de calidad (consulte el Capítulo 14, “Más allá de la funcionalidad”). Escuche las palabras que describen las características deseables del sistema: rápido, fácil, fácil de usar, confiable, seguro. Deberá trabajar con los usuarios para comprender qué quieren decir con estos términos ambiguos y subjetivos para que pueda escribir objetivos de calidad claros y verificables. Los siguientes ejemplos sugieren cómo podrían sonar los atributos de calidad cuando los describen los usuarios:

- “El software móvil debe responder rápidamente a los comandos táctiles”.
- “El mecanismo del carrito de compras tiene que ser simple de usar para que mis nuevos clientes no abandonen la compra”.

**Requisitos de la interfaz externa** Los requisitos de esta categoría describen las conexiones entre su sistema y el resto del universo. La plantilla SRS en el Capítulo 10, "Documentación de los requisitos", incluye secciones para interfaces para usuarios, hardware y otros sistemas de software. Frases como “Debe leer las señales de . . .”, “Debe enviar mensajes a . . .”, “Debe poder leer archivos en <formato>” y “Los elementos de la interfaz de usuario deben cumplir con <un estándar>” indican que el cliente está describiendo un requisito de interfaz externa. Los siguientes son algunos ejemplos:

- “El sistema de ejecución de la fabricación debe controlar el clasificador de obleas”.
- “La aplicación móvil debe enviar la imagen del cheque al banco después de fotografiar el cheque que estoy depositando.

**Restricciones** Las restricciones de diseño e implementación restringen legítimamente las opciones disponibles para el desarrollador (consulte el Capítulo 14). Los dispositivos con software incorporado a menudo deben respetar restricciones físicas como el tamaño, el peso y las conexiones de la interfaz. Las frases que indican que el cliente está describiendo una restricción de diseño o implementación incluyen: "Debe estar escrito en <un lenguaje de programación específico>", "No puede exceder <algún límite>" y "Debe usar <un control

de interfaz de usuario específico>". Los siguientes son ejemplos de restricciones que un cliente puede presentar:

- "Los archivos enviados electrónicamente no pueden exceder los 10 MB de tamaño".
- "El navegador debe usar un cifrado de 256 bits para todas las transacciones seguras".

Al igual que con los requisitos funcionales, no se limite a transcribir la declaración del usuario de una restricción. Pregunte por qué existe la restricción, confirme su validez y registre la justificación para incluirla como requisito.

**Requisitos de datos** Los clientes presentan un requisito de datos cada vez que describen el formato, el tipo de datos, los valores permitidos o el valor predeterminado para un elemento de datos; la composición de una estructura de datos comerciales compleja; o un informe que se generará (consulte el Capítulo 13, "Especificación de los requisitos de datos"). Algunos ejemplos de requisitos de datos son los siguientes:

- "El código postal tiene cinco dígitos, seguidos de un guion opcional y cuatro dígitos cuyo valor predeterminado es 0000".
- "Un pedido consta de la identidad del cliente, la información de envío y uno o más productos, cada uno de los cuales incluye el número de producto, el número de unidades, el precio unitario y el precio total.

**Ideas de solución** Muchos "requisitos" de los usuarios son realmente ideas de solución. Alguien que describe una forma específica de interactuar con el sistema para realizar alguna acción está sugiriendo una solución. El analista de negocios necesita sondear debajo de la superficie de una idea de solución para llegar al requisito real. Preguntar repetidamente "por qué" el usuario necesita que funcione de esta manera probablemente revelará la verdadera necesidad (Wiegers 2006). Por ejemplo, las contraseñas son solo una de varias formas posibles de implementar un requisito de seguridad. Otros dos ejemplos de ideas de solución son los siguientes:

- "Luego selecciono el estado donde quiero enviar el paquete de una lista desplegable".
- "El teléfono debe permitir que el usuario pase el dedo para navegar entre las pantallas".

En el primer ejemplo, la frase de una lista desplegable indica que esta es una idea de solución porque describe un control de interfaz de usuario específico. El BA prudente preguntará: "¿Por qué de una lista desplegable?" Si el usuario responde: "Esa me pareció una buena manera de hacerlo", entonces el requisito real es algo así como: "El sistema le permitirá al usuario especificar el estado donde desea enviar el paquete". Pero tal vez el usuario diga: "Hacemos lo mismo en varios otros lugares y quiero que sea consistente. Además, la lista desplegable evita que el usuario ingrese datos no válidos". Estas son razones legítimas para especificar una solución específica. Sin embargo, reconozca que incorporar una solución en un requisito impone una restricción de diseño sobre ese

requisito: limita el requisito a implementarse de una sola manera. Esto no es necesariamente incorrecto o malo; solo asegúrese de que la restricción esté ahí por una buena razón.

Clasificar la entrada del cliente es solo el comienzo del proceso para crear especificaciones de requisitos. Todavía necesita ensamblar la información en colecciones de requisitos bien organizadas y claramente definidas. A medida que trabaja con la información, elabore requisitos individuales claros y guárdelos en las secciones apropiadas de las plantillas de documentos o repositorio del equipo. Realice pases adicionales a través de esta información para asegurarse de que cada declaración demuestre las características de los requisitos de alta calidad como se describe en el Capítulo 11. A medida que procesa sus notas de “elicitación”, marque los elementos como completos mientras los almacena en el lugar correcto.

## ¿Cómo sabes cuándo has terminado?

Ninguna señal simple indicará cuándo ha completado la obtención de requisitos. De hecho, nunca terminará por completo, especialmente si está implementando deliberadamente un sistema de forma incremental, como en proyectos ágiles. A medida que las personas reflexionan en la ducha cada mañana y hablan con sus colegas, generarán ideas para requisitos adicionales y querrán cambiar algunos de los que ya tienen. Las siguientes señales sugieren que está llegando al punto de rendimientos decrecientes en la obtención de requisitos, al menos por ahora. **Tal vez hayas terminado si:**

- Los usuarios no pueden pensar en más casos de uso o historias de usuarios. Los usuarios tienden a identificar los requisitos de los usuarios en una secuencia de importancia decreciente.
- Los usuarios proponen nuevos escenarios, pero no conducen a nuevos requisitos funcionales. Un caso de uso "nuevo" podría ser realmente un flujo alternativo para un caso de uso que ya capturó.
- Los usuarios repiten temas que ya cubrieron en debates anteriores.
- Las nuevas características sugeridas, los requisitos del usuario o los requisitos funcionales se consideran fuera del alcance.
- Los nuevos requisitos propuestos son todos de baja prioridad.
- Los usuarios proponen capacidades que podrían incluirse “en algún momento de la vida útil del producto” en lugar de “en el producto específico del que estamos hablando en este momento”.
- Los desarrolladores y probadores que revisan los requisitos para un área plantean pocas preguntas.

Combinar la entrada de requisitos de numerosos usuarios es difícil sin utilizar un esquema de organización estructurado, como casos de uso o las secciones en una plantilla de SRS. **A pesar de sus mejores esfuerzos para descubrir todos los requisitos, no lo hará, así que espere hacer cambios a medida que avanza la construcción.** Recuerde, su objetivo es acumular una comprensión compartida de los requisitos que sea lo suficientemente buena como para permitir que la construcción de la próxima versión o incremento continúe con un nivel de riesgo aceptable.

## Algunas precauciones sobre la “elicitación”

**Equilibre la representación de las partes interesadas** Recolectar información de muy pocos representantes o escuchar la voz del cliente más ruidoso y con más opiniones es un problema. Puede llevar a pasar por alto requisitos que son importantes para ciertas clases de usuarios o a incluir requisitos que no representan las necesidades de la mayoría de los usuarios. El mejor equilibrio involucra a algunos campeones de productos que pueden hablar en nombre de sus respectivas clases de usuarios, con cada campeón respaldado por otros representantes de la misma clase de usuario.

**Defina el alcance adecuadamente** Durante la obtención de requisitos, es posible que encuentre que el alcance del proyecto no está definido correctamente, ya sea que sea demasiado grande o demasiado pequeño. Si el alcance es demasiado grande, acumulará más requisitos de los necesarios para ofrecer un valor comercial y de cliente adecuado, y el proceso de “elicitación” se alargará. Si el alcance del proyecto es demasiado pequeño, los clientes presentarán necesidades que son claramente importantes, pero que se encuentran claramente más allá del alcance limitado actualmente establecido para el proyecto. El alcance actual podría ser demasiado pequeño para producir un producto satisfactorio. Por lo tanto, obtener los requisitos del usuario puede llevar a modificar la visión del producto o el alcance del proyecto.

**Evite el argumento de requisitos versus diseño.** A menudo se afirma que los requisitos se refieren a lo que el sistema tiene que hacer, mientras que cómo se implementará la solución es el ámbito del diseño. Aunque atractivamente conciso, esto es una simplificación excesiva. De hecho, la obtención de requisitos debería centrarse en el qué, pero hay un área gris, no una línea nítida, entre el análisis y el diseño (Wiegers 2006). Los cómo hipotéticos ayudan a aclarar y refinar la comprensión de lo que necesitan los usuarios. Los modelos de análisis, los bocetos de pantalla y los prototipos ayudan a hacer más tangibles las necesidades expresadas durante la “elicitación” y a revelar errores y omisiones. Deje en claro a los usuarios que estas pantallas y prototipos son solo ilustrativos, no necesariamente la solución definitiva.

**Investigación dentro de la razón** La necesidad de realizar una investigación exploratoria a veces interrumpe la “elicitación”. Surge una idea o una sugerencia, pero se requiere una investigación exhaustiva para evaluar si se debe considerar para el producto. Trate estas exploraciones de viabilidad o valor como tareas de proyecto por derecho propio. La creación de prototipos es una forma de explorar estos problemas. Si su proyecto requiere una investigación exhaustiva, use un enfoque de desarrollo incremental para explorar los requisitos en porciones pequeñas y de bajo riesgo.

## Requisitos supuestos e implícitos

Nunca documentará el 100 por ciento de los requisitos de un sistema. Pero los requisitos que no especifica plantean el riesgo de que el proyecto pueda ofrecer una solución diferente de lo que esperan las partes interesadas. Se suponen dos posibles culpables detrás de las expectativas perdidas y los requisitos implícitos:

- Los requisitos asumidos son aquellos que la gente espera sin haberlos expresado explícitamente. Lo que asume como obvio puede no ser lo mismo que las suposiciones que hacen varios desarrolladores.
- Los requisitos implícitos son necesarios debido a otro requisito, pero no se establecen explícitamente. Los desarrolladores no pueden implementar funciones que no conocen.

Para reducir estos riesgos, intente identificar las brechas de conocimiento que esperan ser llenadas con requisitos implícitos y asumidos. Pregunte: "¿Qué estamos asumiendo?" durante las sesiones de "elicitación" para tratar de sacar a la superficie esos pensamientos ocultos. Si se encuentra con una suposición durante las discusiones de requisitos, regístrela y confirme su validez. Las personas a menudo asumen que las cosas tienen que ser como siempre han sido porque están muy familiarizadas con un sistema o proceso comercial existente. Si está desarrollando un sistema de reemplazo, revise las funciones del sistema anterior para determinar si realmente se requieren en el reemplazo.

Para identificar los requisitos implícitos, estudie los resultados de las sesiones iniciales de "elicitación" para identificar áreas incompletas. ¿Es necesario desarrollar un requisito vago de alto nivel para que todas las partes interesadas lo entiendan? ¿Un requisito que podría ser parte de un conjunto lógico (por ejemplo, guardar un formulario web incompleto) carece de su contraparte (recuperar un formulario guardado para seguir trabajando)? Es posible que deba volver a entrevistar a algunas de las mismas partes interesadas para que busquen los requisitos que faltan (Rose-Coutré 2007). Además, piense en nuevas partes interesadas que conozcan el tema y puedan detectar brechas.

Lea entre líneas para identificar las funciones o características que los clientes esperan que se incluyan sin haberlo dicho. Haga preguntas libres de contexto, preguntas abiertas y de alto nivel que obtengan información sobre las características globales tanto del problema comercial como de la posible solución (Gause y Weinberg 1989). La respuesta del cliente a preguntas como "¿Qué tipo de precisión se requiere en el producto?" o "¿Puedes ayudarme a entender por qué no estás de acuerdo con la respuesta de Miguel?". Puede conducir a conocimientos que las preguntas con respuestas estándar de sí/no o A/B/C no lo hacen.



## Encontrar los requisitos que faltan

Los requisitos faltantes constituyen un tipo común de defecto de requisito. ¡Los requisitos faltantes son difíciles de detectar porque son invisibles! Las siguientes técnicas le ayudarán a detectar previamente requisitos no descubiertos:

- Descomponga los requisitos de alto nivel con suficiente detalle para revelar exactamente lo que se solicita. Un requisito vago de alto nivel que deja mucho a la interpretación del lector generará una brecha entre lo que el solicitante tiene en mente y lo que construye el desarrollador.

- Asegúrese de que todas las clases de usuarios hayan proporcionado información. Asegúrese de que cada requisito de usuario tenga al menos una clase de usuario identificada que recibirá valor del requisito.

- Rastree los requisitos del sistema, los requisitos del usuario, las listas de respuestas a eventos y las reglas de negocios hasta sus requisitos funcionales correspondientes para asegurarse de que se haya derivado toda la funcionalidad necesaria.

- Verifique los valores límite para los requisitos faltantes. Suponga que un requisito establece: "Si el precio del pedido es inferior a \$100, el cargo de envío es de \$5,95" y otro dice: "Si el precio del pedido es superior a \$100, el cargo de envío es el 6 por ciento del pedido total". Pero, ¿cuál es el costo de envío para un pedido con un precio de exactamente \$100? No está especificado, por lo que falta un requisito, o al menos está mal escrito.

- Representar la información de requisitos en más de una forma. Es difícil leer una gran cantidad de texto y notar el elemento que está ausente. Algunos modelos de análisis representan visualmente los requisitos con un alto nivel de abstracción: el bosque, no los árboles. Podrías estudiar un modelo y darte cuenta de que debería haber una flecha de un cuadro a otro; esa flecha faltante representa un requisito faltante. Los modelos de análisis se describen en el Capítulo 12, "Una imagen vale 1024 palabras".

- Los conjuntos de requisitos con lógica booleana compleja (Y, O y NO) suelen estar incompletos. Si una combinación de condiciones lógicas no tiene un requisito funcional correspondiente, el desarrollador debe deducir qué debe hacer el sistema o buscar una respuesta. Con frecuencia se pasan por alto las condiciones "else". Representar la lógica compleja utilizando tablas de decisión o árboles de decisión para cubrir todas las situaciones posibles, como se describe en el Capítulo 12.

- Cree una lista de verificación de áreas funcionales comunes a considerar para sus proyectos. Los ejemplos incluyen el registro de errores, la copia de seguridad y la restauración, la seguridad de acceso, los informes, la impresión, las capacidades de vista

previa y la configuración de las preferencias del usuario. Compare periódicamente esta lista con las funciones que ya ha especificado para buscar lagunas.

■ Un modelo de datos puede revelar la funcionalidad faltante. Todas las entidades de datos que manipulará el sistema deben tener la funcionalidad correspondiente para crearlas, leerlas desde una fuente externa, actualizar los valores actuales y/o eliminarlas. El acrónimo CRUD se usa a menudo para referirse a estas cuatro operaciones comunes. Asegúrese de que puede identificar la funcionalidad en su aplicación para realizar estas operaciones en todas sus entidades que las necesitan (consulte el Capítulo 13).

**Trampa** Tenga cuidado con la temida parálisis del análisis, dedicando demasiado tiempo a la “elicitación” de requisitos en un intento de evitar perder alguno.

Es probable que nunca descubra todos los requisitos para su producto, pero casi todos los equipos de software pueden hacer un mejor trabajo de obtención de requisitos aplicando las prácticas descritas en este capítulo.

### Próximos pasos

■ Piense en los requisitos que se encontraron tarde en su último proyecto. ¿Por qué se pasaron por alto durante la “elicitación”? ¿Cómo podrías haber descubierto cada uno de estos requisitos antes? ¿Cuánto habría valido eso para su organización?

■ Seleccione una parte de cualquier entrada documentada del cliente en su proyecto o una sección del SRS. Clasifique cada elemento en ese fragmento de requisitos en las categorías que se muestran en la Figura 7-7. Si encuentra artículos que se organizaron incorrectamente, muévalos al lugar correcto en su documentación de requisitos.

■ Enumere las técnicas de obtención de requisitos utilizadas en su proyecto anterior o actual. ¿Cuáles funcionaron bien? ¿Por qué? ¿Cuáles no funcionaron tan bien? ¿Por qué no? Identifique las técnicas de “elicitación” que cree que funcionarían mejor y decida cómo las aplicaría la próxima vez. Identifique cualquier barrera que pueda encontrar para hacer que esas técnicas funcionen y haga una lluvia de ideas sobre cómo superar esas barreras.