

# Los Niveles de Granularidad del Requerimiento Funcional

Por Guilherme Simões y Carlos Vázquez

● **Como la mayoría de los lectores de SG saben**, todo software posee dos dimensiones de requerimientos: funcional y no funcional. Los requerimientos funcionales (RF) describen el comportamiento del software para proporcionar tareas y servicios a sus usuarios; estos tipos de requerimientos están relacionados a lo que el software debe hacer. Los requerimientos no funcionales (RNF) están relacionados al cómo las funcionalidades del software serán entregadas a sus usuarios y pueden incluir aspectos de calidad, técnicos, ambientales y organizacionales.

Teniendo en cuenta un sistema de cajero automático de un banco se podría suponer los siguientes ejemplos para los dos tipos de requerimientos:

- RF: Consultar saldo de la cuenta corriente (el qué).
- RNF: Toda transacción debe ser completada en un máximo de 30 segundos (el cómo).

Aunque no sea una regla, el RF se acostumbra manifestar de manera explícita en el software, es un comportamiento que se puede determinar claramente en qué parte del software está. El RNF, por lo contrario, se acostumbra manifestar de manera general, se aplica a todo o casi todo el software.

Para la Ingeniería de Requerimientos los dos tipos son igualmente relevantes: ambos deben ser identificados, analizados, especificados, modelados y gestionados. Sin embargo, hay algunas diferencias entre estos dos tipos de requerimientos que valen la pena comentar por separado. Este artículo se enfocará en los RF, con el objetivo de presentar sus niveles de granularidad y su importancia en el trabajo de la Ingeniería de Requerimientos.

## LOS NIVELES DE GRANULARIDAD

Continuando con el ejemplo del sistema de cajero automático, las siguientes descripciones podrían estar presentes en su especificación de requerimientos:

1. Realizar transacciones con la cuenta corriente.
2. Transferir una cifra de una cuenta corriente para otra cuenta corriente.

3. Validar la tarjeta y la contraseña del cliente.
4. Garantizar que la suma de todas las transacciones del cliente en el día no puede ser superior a \$5,000.

Aunque estos cuatro ejemplos sean casos válidos de RF, es posible percibir que el nivel de detalle (o granularidad) entre ellos es distinto. Lo más frecuente es que una especificación presente los requerimientos en distintos niveles de granularidad.

El nivel de granularidad es la mayor o menor amplitud de la descripción del comportamiento esperado para el software en una especificación funcional. Esto está relacionado al tipo de objetivo asociado al requerimiento. La Figura 1 ilustra la relación entre estos objetivos y el nivel de granularidad, utilizando una clasificación de tres niveles (agregado, usuario, subfunción) propuesta por Cockburn (2000) para casos de uso y generalizada por los autores para requerimientos funcionales. Esta clasificación no significa que el proceso de la ingeniería de requerimientos deba utilizar una estrategia de desarrollo por descomposición funcional.

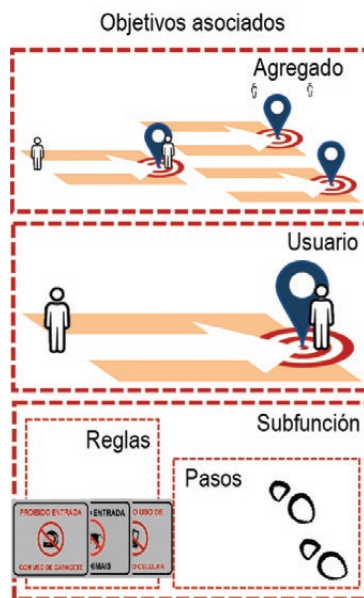


Figura 1. Niveles de granularidad de los RF.

Guilherme Siqueira Simões ([guilherme.simoes@fattocs.com](mailto:guilherme.simoes@fattocs.com)) y Carlos Eduardo Vazquez ([carlos.vazquez@fattocs.com](mailto:carlos.vazquez@fattocs.com)) son autores de dos libros en portugués: uno sobre el Análisis de Puntos de Función y otro sobre la Ingeniería de Requerimientos. Ambos tienen más de 20 años de experiencia con desarrollo de software y son socios de Fatto Consultoria y Sistemas ([www.fattocs.com](http://www.fattocs.com)), que oferta servicios de consultoría y entrenamiento en las disciplinas de medición, estimación y requerimientos de software.

## NIVEL DE USUARIO

El requerimiento a nivel de objetivo de usuario es aquel que:

- Abarca una única tarea bajo la responsabilidad de un único individuo;
- Es realizado en el momento en que el usuario posee todo lo que es necesario para que la tarea sea completada hasta el límite de su responsabilidad en el flujo operativo donde está contenida.

Al final de la tarea el usuario alcanza su propósito, está satisfecho y no hay nada más que necesite hacer. Una vez que el requerimiento fue completado, todo lo que debería de realizarse para tratar un evento externo fue realizado. Esta tarea es casi siempre parte de un proceso de negocio que puede tener un flujo operativo más amplio y por esto puede aún no estar completa. Sin embargo, la perspectiva relevante en este caso no es del proceso de negocio, es de la tarea.

En general, si un trabajo involucra más de un individuo es porque hay más de una tarea presente en el contexto. Hay excepciones como un retiro en la cuenta corriente en la sucursal del banco donde dos individuos participan: la caja del banco que solicita e informa datos para el retiro y el cliente que informa la contraseña.

Esta es la granularidad del ejemplo 2 (“Transferir una cifra de una cuenta corriente para otra cuenta corriente.”). Es una única tarea (seguramente compuesta por varios pasos y reglas), bajo la responsabilidad de un único individuo que al final de todos los pasos está satisfecho con el objetivo alcanzado: una cifra fue transferida a otra cuenta corriente.

## NIVEL AGREGADO

El requerimiento en este nivel agrega varios requerimientos a nivel de objetivo de usuario en una única especificación de más alto nivel. Tanto más alto sea su nivel, más generales son sus objetivos y para que un objetivo de nivel más alto sea alcanzado, otros objetivos de nivel más bajo deben ser alcanzados primero.

Este tipo de requerimiento está relacionado a objetivos más generales y su amplitud está asociada a objetivos colaborativos o asociados a procesos de negocio de alto nivel. No es relativo a una única tarea, por lo contrario, agrega un conjunto de tareas de uno o más usuarios. Esta es la granularidad del ejemplo 1 (“Realizar transacciones con la cuenta corriente.”).

¿Cuáles son las tareas asociadas a este tipo de requerimiento? Tal vez sean obvias para los lectores de la especificación (los interesados) o tal vez aún no sean conocidas. Sin embargo, es sabido que hay actividades a realizar para levantar (refinar) este requerimiento. Lo que importa es que ya está claro que este requerimiento está dentro del alcance del software a ser desarrollado.

Sin embargo, algunos RF de este tipo poseen patrones que eliminan la necesidad de proporcionar más detalles. Un ejemplo clásico son los formularios CRUD (Create, Read, Update, Delete) para que el usuario pueda mantener datos por medio del software. Es muy común que esto sea expresado como: “El sistema debe mantener productos.”. Hay un conocimiento tácito entre los interesados que el verbo “mantener” agrega las tareas del CRUD. Por lo tanto queda claro a todos los lectores que el software debe permitir al usuario realizar las siguientes tareas: agregar, modificar, eliminar y consultar datos de producto.

## NIVEL DE SUBFUNCIÓN

Estos requerimientos son pedazos de requerimientos con objetivo de usuario; están relacionados a un conjunto de pasos o a reglas de una o más tareas de los usuarios.

El requerimiento en nivel de subfunción que representa un conjunto de pasos describe el cambio de datos en los dos sentidos entre el usuario y el software; y entre el software y los requerimientos de almacenamiento. Este es el caso del ejemplo 3 (“Validar la tarjeta y la contraseña del cliente.”). Cada tipo de transacción que utiliza la cuenta corriente (ej.: retiro, transferencia, pago, etc.) exige el mismo conjunto de pasos descrito por el ejemplo 3, que se podría suponer como:

- Verificar si la tarjeta es válida.
- Verificar si la transacción elegida es compatible con el tipo de tarjeta.
- Verificar si la contraseña informada es correcta.
- Incrementar el control de errores de contraseña en caso que la contraseña informada sea incorrecta.
- Cambiar a cero el control de errores de contraseña en caso que la contraseña informada sea correcta.

Validar la tarjeta y contraseña del cliente no es el objetivo del cliente al utilizar el sistema de cajero automático, sin embargo son pasos necesarios y intermediarios para alcanzar su objetivo: por ejemplo, hacer un retiro.

El requerimiento en el nivel de subfunción que está relacionado a reglas, en general se vincula a las leyes que gobiernan el negocio y describen de manera que complementan los procesos de negocio. También llamadas muchas veces como reglas de negocio. La regla puede describir políticas corporativas, reglamentos gubernamentales o estándares de la industria por los cuales el software debe estar subordinado.

Este es el caso del ejemplo 4 (“Garantizar que la suma de todas las transacciones del cliente en el día no puede ser superior a \$5,000.”). Las reglas de negocio muchas veces son compartidas entre varios RF, hasta entre distintos productos de software en la empresa.

## IMPORTANCIA DE LA GRANULARIDAD

En una visión simplificada se puede decir que hay dos momentos clave donde una especificación de requerimientos es necesaria:

- A. Proporcionar una visión amplia del software (y tal vez aún no profunda).
- B. Proporcionar una visión profunda del software (de una parte o del todo).

El caso A es muy común en etapas tempranas de proyectos. El objetivo en este momento es primero obtener una visión amplia del alcance, por ejemplo: para planificar el proyecto, estimar un orden de magnitud de costo o plazo, o hacer un análisis de factibilidad. Un documento de visión es un ejemplo de documento que cumple este propósito. Especificar los RF en el nivel agregado es suficiente y más adecuado a este caso. El costo para obtener información y especificar en un nivel de granularidad más bajo puede significar desperdicio de esfuerzo para detallar lo que no es necesario.

Entonces el documento de requerimientos para el propósito A, tendrá la mayoría de los RF especificados en el nivel agregado. Es posible también que haya especificaciones en el nivel de usuario o subfunción, esto no es un error. A veces puede ser interesante especificar los RF más críticos (no todos o la mayoría) en un nivel más detallado, ya sea para facilitar la comprensión del alcance por el interesado o permitir una estimación con menos incertidumbre.

El principal propósito para especificar requerimientos en niveles agregados es establecer áreas de proceso objeto de informatización y/o automatización para que las necesidades de negocio queden resueltas.

Si el objetivo de la especificación es proporcionar una visión profunda de parte o de todo el software, por ejemplo para iniciar su desarrollo, la existencia de RF en el nivel agregado puede significar que varias decisiones sobre el alcance todavía siguen pendientes. Es decir, hay trabajo de levantamiento que debe ser hecho.

Para el propósito B, el nivel de granularidad más adecuado es del objetivo del usuario pues es el único que proporciona una definición de alcance del software de manera inequívoca; no hay dudas sobre lo que es abarcado. Esto es también el único nivel de descripción de procesos que puede ser estandarizado y consecuentemente es el nivel utilizado por todos los métodos de medición de tamaño funcional adherentes a la norma ISO/IEC 14143.

Consecuentemente, para el propósito B, lo más común es que un documento de requerimientos tenga los RF especificados en su mayoría con el objetivo de usuario. RF en el nivel agregado puede estar presente sólo cuando es el caso de que esté claro a todos los interesados que tareas son abarcadas por este RF.

La definición de un requerimiento en el nivel de subfunción es adecuada cuando el comportamiento especificado es común y compartido por varias otras tareas que el software entrega a los usuarios. Esto hace con que la especificación de requerimientos sea más fácil de modificar en respuesta a cambios, pues disminuye la redundancia de información al evitar describir el mismo conjunto de pasos o reglas más de una vez. También ayuda a alcanzar una especificación más consistente.

## CONCLUSIÓN

Aunque el concepto de los niveles de granularidad del requerimiento funcional sea sencillo, en términos prácticos se percibe que muchos ingenieros de requerimientos no están atentos a esto cuando elaboran las especificaciones. Tener en cuenta los niveles de granularidad ayuda a definir el esfuerzo adecuado para ser dedicado a la elaboración de la especificación y también a alcanzar una especificación de requerimientos de mejor calidad.

El estándar 830 de la IEEE presenta los siguientes criterios para evaluar la calidad de una especificación de requerimientos: correcta, completa, clara (no ambigua), consistente, modificable, priorizada, verificable, trazable.

Siendo el propósito de la especificación proporcionar una visión profunda del alcance, estar atento al nivel de granularidad del RF permite que:

- Se descubra que la especificación puede no estar completa cuando hay RF agregados.
- Se proporcione la comprensión del alcance sin ninguna ambigüedad con respeto a las tareas que el software entregará cuando se enfoca el RF con objetivo de usuario.
- Se genere un documento de requerimientos más fácil de mantener y consecuentemente más consistente al especificar los RF adecuados con objetivo de subfunción. ☺

## Referencias

- [1] [A. Cockburn. \*Writing Effective Use Cases\*. Addison-Wesley, 2000.](#)
- [2] [COSMIC Measurement Manual: 4.0.1. Common Software Measurement International Consortium, 2015.](#)
- [3] [IEEE Std. 830-1998: IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.](#)
- [4] [Function Point Counting Practices Manual. Release 4.3.1. IFPUG, 2010.](#)
- [5] [C. Vazquez, G. Simões & R. Albert. \*Análise de Pontos de Função, Medição, Estimativas e Gerenciamento de Projetos de Software\*. Saraiva, 2013.](#)
- [6] [C. Vazquez, G. Simões. \*Engenharia de Requisitos: Software Orientado ao Negócio\*. Brasport, 2016.](#)