

Ejemplos y buenas prácticas para descomponer historias de usuario en tareas (parte 1 de 2)

[Deja un comentario](#) / [agil](#) / [Por jgarzas](#) / 3 mayo, 2012

Aun incluso después de leer el [User Stories Applied](#) de Cohen, parece que el tema de cómo, en un proyecto ágil, descomponer historias de usuario en tareas (te recomiendo aquí este [post sobre historias de usuario](#)) para muchos [no queda del todo claro](#). Y este es uno de esos problemas que siempre salen cuando llega el momento de trabajar con historias de usuario: la descomposición en tareas.

Para aclarar este tema, en el [libro sobre gestión ágil de proyectos que estamos escribiendo](#), introducimos un conjunto de ejemplos y buenas prácticas que se recomiendan aplicar a la hora de descomponer historias de usuario en tareas. Afortunadamente, aparte de nuestra experiencia, también hemos podido contar con algunos buenos textos que tratan este tema (por ejemplo [este](#) y [este](#)).

Os dejo, en esta serie de dos post (la parte 2 la publicaré el lunes), primero un extracto de ejemplos y luego unas buenas prácticas para descomponer historias de usuario en tareas.

Ejemplos que te pueden ayudar a descomponer historias de usuario en tareas:

- **Diseño de la historia de usuario:** especialmente para generar discusión sobre cómo se va a implementar la historia de usuario.
- **Implementación de una historia:** esta tarea puede incluir la definición de las interfaces y los métodos necesarios para cubrir la necesidad expresada en la historia. Pueden existir múltiples tareas de este tipo.
- **Pruebas unitarias asociadas a la historia (TDD):** esta tarea debiera ser obligatoria para cada historia.
- **Pruebas de aceptación asociadas a la historia:** tarea definida para desarrollar las pruebas de aceptación automáticas que facilitarán la validación de la historia por parte del cliente y/o usuario.
- **Requisitos no funcionales:** para cada historia, se deben tener en cuenta requisitos de seguridad, rendimiento, escalabilidad, etc. Estos atributos de calidad pueden dar lugar a nuevas tareas.
- **Revisiones de código:** el código siempre debe haber sido revisado, convirtiéndose la revisión en otra tarea.
- **Refactorización del código (te recomiendo aquí leer este [post sobre qué es la refactorización](#)):** Esta tarea se debe incluir para cada historia con el fin de evitar una complejidad excesiva del código.
- **Emular interfaces:** en el caso de que se demore en exceso la implementación de una interfaz, éstas se deben emular con el fin de comenzar a trabajar.
- **Pruebas exploratorias:** pruebas ad-hoc para una historia.
- **Corrección de errores:** dependiendo de la historia, se debe reservar tiempo para la resolución de errores o incidencias.
- **Verificación de errores:** dependiendo de la historia, se debe reservar tiempo para verificar la corrección de los errores.
- **Demo:** demostración interna al equipo de la historia. Esta pequeña demostración se suele realizar una vez finalizada la historia en la reunión diaria.
- **Actualizar la wiki o el repositorio de documentos:** con el diseño y los resultados de la historia.
- **Documentación de usuario final:** la generación de los manuales de usuario, instalación, etc. también debe estar ubicada en alguna tarea.

Hay que tener en cuenta que el anterior es solamente un listado de ejemplo, y que dependiendo de la historia de usuario habrá elementos de esta lista que pueden aplicar o no. A su vez, en los proyectos ágiles pueden surgir otros tipos de tareas que no se encuadren en esta lista y que sean igualmente válidos.

¿Alguien tiene algún ejemplo más, o recomendación, sobre descomponer historias de usuario en tareas?

Ejemplos y buenas prácticas para descomponer historias de usuario en tareas (parte 2 de 2)

[Deja un comentario](#) / [General](#) / [Por jgarzas](#) / 7 mayo, 2012

En la primera parte de esta serie de dos post sobre cómo descomponer historias de usuario en tareas (te recomiendo aquí este [post sobre historias de usuario](#)), vimos algunos ejemplos típicos de tareas. En esta segunda parte os dejo algunas recomendaciones y buenas prácticas a tener en cuenta a la hora de hacer esa descomposición.

Buenas prácticas para descomponer historias de usuario en tareas

- **A la hora de descomponer historias de usuario en tareas, intenta que el tamaño de las tareas sea de entre medio día hasta un máximo de 3 o 4 días de trabajo de un solo miembro del equipo.** Tareas más pequeñas suelen conllevar grandes pérdidas de tiempo a la hora de gestionarlas. Por otro lado, las tareas de más de 3 o 4 días de trabajo, se pueden dividir en otras tareas, si es posible, con el fin de no ocupar demasiado tiempo y ser completadas de manera eficiente.
- **Crear tareas que una vez completadas generen un producto entregable.** Por ejemplo, en vez de tareas como "Construir la interfaz de usuario", "Construir la lógica de negocio" o "Construir la capa de persistencia", una división que se recomendaría para este tipo de funcionalidad sería "Implementar el módulo de inserción de un nuevo usuario", "Implementar el módulo de actualización de un usuario", o "Implementar el módulo de eliminación de un usuario". Así cuando se complete una tarea se podrá generar un producto entregable. A su vez estas tareas no dependen de la finalización de otras para ser probadas.
- **No dedicar excesivo tiempo a estudiar todos los detalles de cada tarea.** Suele ocurrir que una vez definida la tarea, el siguiente paso sea su estimación, y que para ello el equipo quiera definir y conocer todos los detalles posibles de la misma. Aunque, efectivamente, este análisis ayuda a realizar estimaciones más precisas, puede demorar en gran medida el proceso de división de las historias de usuario en tareas.
- **En el conjunto de tareas, a la hora de descomponer historias de usuario en tareas, deben estar incluidas tareas de pruebas y su posible automatización.**