



SCRUM

Febrero 2010

Scrum: Desarrollado y mantenido por Ken Schwaber y Jeff Sutherland

Agradecimientos

General

Scrum se basa en buenas prácticas aceptadas por la industria, usadas y probadas durante décadas. Se establece como una teoría empírica de proceso. Como Jim Coplien comentó una vez a Jeff, "A todo el mundo le gustará Scrum; es lo que ya hacemos cuando estamos entre la espada y la pared."

Las personas

De las miles de personas que han contribuido a Scrum, debemos destacar los que jugaron un papel decisivo en sus primeros diez años. En primer lugar estuvo Jeff Sutherland, trabajando con Jeff McKenna, y Ken Schwaber con Mike Smith y Chris Martin. Scrum fue presentado formalmente por primera vez y publicado en OOPSLA en 1995. Durante los siguientes cinco años, Mike Beadle y Martine Devos hicieron contribuciones significativas. Y luego todos los demás, sin cuya ayuda, Scrum no habría sido perfeccionado hasta lo que es hoy en día.

Historia

La historia de Scrum se puede considerar larga en el mundo del desarrollo de software. En honor a los primeros lugares en los que fue probado y refinado, mencionamos a Individual Inc., Fidelity Investments e IDX (en la actualidad GE Medical).

Traducción

Esta guía ha sido traducida de la versión original en inglés, proporcionada por Ken Schwaber y Jeff Sutherland. Han contribuido a la traducción: Marcelo R. Lopez Jr., Nestor Stura y Jose Luis Soria.

Propósito

Scrum se ha utilizado para desarrollar productos complejos desde principios de los '90. Esta guía describe cómo utilizar Scrum para desarrollar productos. Scrum no es un proceso o una técnica para desarrollar o crear productos, sino que es un marco en el que se pueden emplear diversos procesos y técnicas. El papel de Scrum es hacer aflorar la eficacia relativa de las prácticas de desarrollo empleadas por usted, para que pueda mejorarlas, **a la vez que proporciona un marco dentro del cual se pueden desarrollar productos complejos.**

Teoría de Scrum

Scrum, que se basa en la teoría del control empírico del procesos, emplea un enfoque iterativo e incremental para optimizar la previsibilidad y controlar los riesgos. Existen tres pilares que sostienen toda implementación del control empírico de procesos.

El primer pilar es la transparencia

La transparencia garantiza que los aspectos del proceso que afectan al resultado, son visibles para aquellos que administran dicho resultado. Estos aspectos no sólo deben ser transparentes, sino también conocidos. Es decir, cuando alguien que inspecciona un proceso cree que algo está hecho, esto debe ser equivalente a su definición de "hecho".

El segundo pilar es la inspección

Se deben inspeccionar con la frecuencia suficiente los diversos aspectos del proceso para que puedan detectarse variaciones inaceptables en el mismo. La frecuencia de inspección debe tener en cuenta que todos los procesos se cambian por el propio acto de inspección. El dilema se presenta cuando la frecuencia de inspección requerida excede la tolerancia del proceso a ser inspeccionado. Afortunadamente, esto parece no aplicar al desarrollo de software. El

otro factor es la habilidad y la diligencia de la gente que inspecciona los resultados del trabajo.

El tercer pilar es la adaptación

Si el inspector determina, a través de la inspección, que uno o más aspectos del proceso están fuera de los límites aceptables, y que el producto resultante será inaceptable, debe ajustar el proceso o el material procesado. El ajuste debe realizarse lo más rápidamente posible para minimizar una desviación mayor.

Hay tres puntos para la inspección y la adaptación en Scrum. La reunión diaria de Scrum se utiliza para inspeccionar el avance hacia la meta de Sprint, y para hacer las adaptaciones que optimicen el valor de la jornada de trabajo del día siguiente. Además, la Revisión de Sprint y las Reuniones de Planificación se utilizan para inspeccionar el progreso hacia el Objetivo (la liberación de una versión) y para hacer las adaptaciones que optimicen el valor del siguiente Sprint. Por último, la Retrospectiva de Sprint se utiliza para revisar el Sprint pasado y determinar qué adaptaciones harán el siguiente Sprint más productivo, satisfactorio y agradable.

Contenido de Scrum

El marco de Scrum se compone de un conjunto de **Equipos Scrum** y sus roles asociados; así como de **Bloques de Tiempo**, **Artefactos**, y **Reglas**.

Los **Equipos Scrum** están diseñados para optimizar la flexibilidad y la productividad, para lo cual, son auto-gestionados, multifuncionales, y trabajan en iteraciones. Cada Equipo Scrum tiene tres roles: 1) el **ScrumMaster**, que es responsable de asegurar que el proceso es comprendido y seguido, 2) el **Propietario del Producto**, que es responsable de maximizar el valor del trabajo realizado por el Equipo Scrum, y 3) el **Equipo**, que hace el trabajo. El equipo está formado por desarrolladores con todos los conocimientos necesarios para convertir los requerimientos del Propietario del Producto en un incremento potencialmente utilizable del producto al final del Sprint.

Scrum emplea bloques de tiempo para crear regularidad. Los elementos de Scrum basados en bloques de tiempo son: la **Reunión de Planificación de la Entrega**, la **Reunión de Planificación del Sprint**, el **Sprint**, el **Scrum Diario**, la **Revisión del Sprint**, y la **Retrospectiva del Sprint**. El corazón de Scrum es un **Sprint**, que es una iteración de un mes de duración o menos. La duración de cada Sprint se mantiene constante a lo largo de todo el esfuerzo de desarrollo. Todos los Sprints utilizan el mismo marco de referencia de Scrum, y proporcionan un incremento de funcionalidad potencialmente utilizable al producto final. Cada Sprint se inicia inmediatamente después del anterior.

Scrum emplea cuatro Artefactos principales. El **Product Backlog** es una lista priorizada de todo lo que podría ser necesario en el producto. El **Sprint Backlog** es una lista de tareas para convertir el Product Backlog correspondiente a un Sprint, en un incremento del producto potencialmente entregable. Un burndown es una medida del backlog restante a través del tiempo. Un **Burndown de Versión** mide el Product Backlog restante durante el tiempo correspondiente a una liberación de una versión. Un **Sprint Burndown** mide los elementos restantes del **Sprint Backlog** en el transcurso de un Sprint.

Consejo

Cuando no existen reglas establecidas, se espera de los usuarios de Scrum que deduzcan qué hacer. No trate de encontrar una solución perfecta, porque el problema por lo general cambia rápidamente. En su lugar, pruebe algo y vea cómo funciona. Los mecanismos de inspección y adaptación de la naturaleza empírica de Scrum le guiarán.

Las **Reglas** sirven de unión para los bloques de tiempo, los roles y los artefactos de Scrum, y se describen en el cuerpo de este documento. Por ejemplo, una regla de Scrum es que sólo los miembros del equipo - la gente comprometida a convertir el Product Backlog en un incremento - pueden hablar durante un Scrum Diario. En las secciones

de "Consejos" que encontrará en este documento se describen formas de aplicar Scrum que no son reglas, sino más bien sugerencias.

Roles de Scrum

El Equipo Scrum consiste en el ScrumMaster, el Propietario del Producto, y el Equipo. A los miembros del equipo Scrum se les llama "cerdos". El Propietario del Producto es el "cerdo" del Product Backlog. El equipo es el "cerdo" del trabajo del Sprint. El ScrumMaster es el "cerdo" del proceso de Scrum. Todo el resto de personas involucradas son "gallinas". Las "gallinas" no pueden decir a los "cerdos" cómo hacer su trabajo. El símil de las "gallinas" y los "cerdos" viene de la siguiente historia:

Consejo

El ScrumMaster trabaja con los clientes y la gerencia para identificar y nombrar al Propietario del Producto. El ScrumMaster explica al Propietario del Producto cómo hacer su trabajo. Se espera que los Propietarios del Producto sepan cómo gestionar para optimizar el valor utilizando Scrum. Si esto no se cumple, la responsabilidad podría recaer en el ScrumMaster.

*"Están juntos una gallina y un cerdo, cuando la gallina dice:"
¡Vamos a abrir un restaurante! "*

El cerdo se lo piensa y dice, "¿Cómo llamaremos al restaurante?"

La gallina, dice, "¡Huevos con Jamón!"

El cerdo dice: "No, gracias, yo estaría comprometido, pero tú solamente estarías involucrada"

El ScrumMaster

El ScrumMaster es responsable de asegurar que el equipo Scrum se adhiere a los valores, prácticas y normas Scrum. El ScrumMaster ayuda a que el Equipo Scrum y la organización adopten Scrum. El ScrumMaster enseña al Equipo Scrum mediante entrenamiento y liderándolo para que sea más productivo y a construya productos de mayor calidad. El ScrumMaster

ayuda a que el Equipo Scrum comprenda y utilice la auto-gestión y a ser multidisciplinar. El ScrumMaster también debe ayudar al Equipo Scrum a entregar lo mejor de sí mismo, en un entorno de organización que posiblemente aún no esté optimizado para el desarrollo de productos complejos. Cuando el ScrumMaster ayuda a realizar estos cambios, esto recibe el nombre de "eliminar impedimentos." El papel del ScrumMaster es el de servidor y líder del equipo de Scrum. Sin embargo, el ScrumMaster no gestiona al Equipo Scrum; el equipo Scrum es auto-gestionado.

Consejo

El ScrumMaster puede ser un miembro del Equipo; por ejemplo, un desarrollador realizando tareas del Sprint. Sin embargo, esto conduce frecuentemente a conflictos cuando el ScrumMaster tiene que elegir entre eliminar obstáculos o realizar las tareas. El ScrumMaster nunca debe ser el Propietario del Producto.

El Propietario del Producto

El Propietario del Producto es la única persona responsable de gestionar el Product Backlog y asegurar el valor del trabajo que el equipo lleva a cabo. Mantiene el Product Backlog y asegura la visibilidad del mismo para todos. Todo el mundo sabe qué elementos tienen la máxima prioridad, por lo que todo el

Consejo

El propietario del producto puede ser un miembro del equipo, que también participe en las tareas de desarrollo. Esta responsabilidad adicional puede disminuir la capacidad del Propietario del Producto de trabajar con los interesados del proyecto o producto. Sin embargo, el propietario del producto no puede ser nunca el ScrumMaster.

mundo sabe en qué se va a trabajar.

El Propietario del Producto es una persona, no un comité. Pueden existir comités que le aconsejen o le influyeran, pero aquellos que quieran cambiar la prioridad de un elemento tendrán que convencerle. Las empresas que adoptan Scrum, pueden encontrarse con el hecho de que Scrum influye en sus métodos para establecer las prioridades y los requisitos a través del tiempo.

Consejo

Para desarrollo comercial, el Propietario del Producto podría ser el gerente de producto. Para los desarrollos o proyectos internos, el Propietario del Producto podría ser el gerente de la función empresarial que se esté automatizando.

Para que el Propietario del Producto tenga éxito, todos en la organización deben respetar sus decisiones. Nadie está autorizado a obligar al Equipo a trabajar bajo un conjunto diferente de prioridades, y a los Equipos no se les permite prestar atención a nadie que diga lo contrario. Las decisiones del Propietario del Producto se hacen visibles en el contenido y la priorización del Product Backlog. Esta visibilidad requiere que el Propietario del Producto dé lo mejor de sí mismo, y hace que el Propietario del Producto sea un rol exigente además de gratificante.

El Equipo

Los Equipos de desarrolladores convierten el Product Backlog en incrementos de funcionalidad potencialmente entregables en cada Sprint. Los Equipos también son multifuncionales; los miembros del equipo deben tener todas las habilidades necesarias para crear un incremento de trabajo. Los miembros del Equipo a menudo tienen habilidades especializadas, como la programación, el control de calidad, el análisis de negocio, la arquitectura, el diseño de la interfaz de usuario o el diseño de bases de datos. Sin embargo, las habilidades que el miembro del Equipo comparte - es decir, la habilidad de tratar con un requisito y convertirlo en un producto utilizable - tienden a ser más importantes que aquellas que no comparte. Las personas que se

niegan a escribir código, ya que son arquitectos o diseñadores, no se ajustan bien a los Equipos. Todo el mundo interviene, incluso si eso requiere aprender nuevas habilidades o recordar las antiguas. No hay títulos en los Equipos, y no hay excepciones a esta regla. Los equipos tampoco contienen sub-Equipos dedicados a áreas particulares, como pruebas o análisis de negocio.

Los Equipos también se auto-organizan. Nadie - ni siquiera el ScrumMaster - dice al Equipo cómo convertir el Product Backlog en incrementos de funcionalidad entregable. El Equipo busca por su cuenta la mejor forma de hacerlo. Cada miembro del equipo aplica su experiencia a todos los problemas. La sinergia resultante mejora la eficiencia global de todo el Equipo, y la eficacia.

El tamaño óptimo para un equipo es de siete personas, más o menos dos. Cuando hay menos de cinco miembros en el Equipo, hay menos interacción y como resultado hay menos aumento de la productividad. Es más, el Equipo puede encontrar limitaciones de habilidad en ciertos momentos del Sprint y no poder entregar una porción entregable del producto. Si hay más de nueve miembros, el problema simplemente es que se necesita demasiada coordinación. Los equipos grandes generan demasiada complejidad como para que pueda ser gestionada por un proceso empírico. Sin embargo, nos hemos encontrado con algunos Equipos exitosos que han superado los límites superior e inferior de este rango de tamaño. Los roles del Propietario del Producto y del ScrumMaster no se computan en este rango a menos que sean también cerdos, que trabajen en las tareas del Sprint Backlog.

La composición del Equipo puede cambiar al final de un Sprint. Cada vez que se cambian los miembros del Equipo, la productividad obtenida de la auto-organización se ve disminuida. Por esta razón se debe tener cuidado al cambiar la composición del equipo.

Bloques de Tiempo

Los Bloques de Tiempo en Scrum son la **Reunión de Planificación de la Entrega**, el **Sprint**, la **Reunión de Planificación del Sprint**, la **Revisión del Sprint**, la **Retrospectiva del Sprint**, y el **Scrum Diario**.

Reunión de Planificación de la Entrega

El propósito de la planificación de la entrega es establecer un plan y unas metas que los Equipos Scrum y el resto de las organizaciones puedan entender y comunicar. La planificación de la entrega responde a las preguntas: "¿Cómo podemos convertir la visión en un producto ganador, de la mejor manera posible? ¿Cómo podemos alcanzar o mejorar la satisfacción del cliente deseada y el Retorno de la Inversión?". El plan de entrega establece el objetivo de la entrega, el Product Backlog de mayor prioridad, los principales riesgos, y las características generales y la funcionalidad que va a contener la entrega. También establece una fecha probable de entrega, y el coste, que debería mantenerse si no cambia nada. La organización puede inspeccionar el avance y hacer cambios a este plan de entrega en cada Sprint.

La planificación de entrega es completamente opcional. Si los Equipos Scrum empiezan a trabajar sin esta reunión, la ausencia de los artefactos generados en ella se revelará en la forma de impedimentos que hay que resolver. El trabajo necesario para resolver cada impedimento se convertirá en un elemento del Product Backlog.

Los productos se construyen utilizando Scrum de forma iterativa, de modo que cada Sprint crea un incremento del producto, empezando por lo más valioso y de mayor riesgo. Más y más Sprints van generando incrementos adicionales del producto. Cada incremento es una parte potencialmente entregable de todo el producto. Cuando se han creado los suficientes incrementos de valor para el producto, de utilidad para sus inversores, la versión del producto es entregada.

La mayoría de las organizaciones ya tienen un proceso de planificación de entregas, y en la mayoría de estos procesos, la mayor parte de la planificación se realiza al comienzo del proceso, y no se modifica a lo largo del tiempo. En la planificación de la entrega en Scrum, se definen un objetivo general y los resultados probables. Esta planificación de entrega por lo general no requiere más de un 15-20% del tiempo consumido por una organización para construir un plan de entrega tradicional. Sin embargo, durante el trabajo en una entrega con Scrum, se realizará planificación sobre la marcha en cada Revisión del Sprint y Reunión de Planificación del Sprint, así como diariamente en cada reunión diaria de Scrum. En total, los esfuerzos de planificación de una entrega de Scrum probablemente consumen muy poco esfuerzo más que los de la planificación de entrega tradicional.

La planificación de una entrega requiere estimar y priorizar los elementos del Product Backlog para dicha entrega. Hay muchas técnicas para hacer esta estimación y priorización, que se encuentran fuera del ámbito de Scrum, pero que son útiles cuando se utilizan en combinación con Scrum.

El Sprint

Un Sprint es una iteración. Los Sprints están limitados en bloques de tiempo. Durante el Sprint, el ScrumMaster asegura que no se realizan cambios que afecten al Objetivo del Sprint. Tanto la composición del Equipo como los objetivos de calidad se mantienen constantes durante todo el Sprint. Los Sprints se componen de: la Reunión de Planificación de Sprint, el trabajo de desarrollo, la Revisión del Sprint, y la Retrospectiva del Sprint. Los Sprints ocurren uno tras otro, sin tiempo entre ellos.

Consejo

Si el Equipo siente que se ha comprometido a demasiado trabajo, se reúne con el Propietario del Producto para eliminar o reducir el alcance del Product Backlog seleccionado para el Sprint. Si el Equipo siente que puede tener tiempo de sobra, puede trabajar con el Propietario del Producto para seleccionar elementos adicionales del Product Backlog.

Un proyecto se utiliza para lograr un resultado; en desarrollo de software, se utiliza para crear un producto o sistema. Cada proyecto consiste en una definición de lo que se va a construir, un plan para construirlo, el trabajo realizado de acuerdo con el plan, y el producto resultante. Cada proyecto tiene un horizonte, es decir, el plazo para el que el plan es bueno.

Consejo

Cuando un Equipo comienza a utilizar Scrum, tener Sprints de dos semanas le permite aprender sin perderse en la incertidumbre. Los Sprints de esta longitud se pueden sincronizar con otros Equipos mediante la entrega de dos incrementos en conjunto.

Si el horizonte es demasiado lejano, la definición puede haber cambiado, pueden haber intervenido demasiadas variables, el riesgo puede ser demasiado grande, etc. Scrum es un marco para un proyecto cuyo horizonte no es superior a un mes, y en el cual hay suficiente complejidad para que un horizonte más largo sea demasiado arriesgado. La previsibilidad del proyecto tiene que ser controlada por lo menos cada mes, y el riesgo de que el proyecto pueda descontrolarse o volverse impredecible, debe contenerse por lo menos cada mes.

Un Sprint puede ser cancelado antes de que el bloque de tiempo del Sprint se haya terminado. Sólo el Propietario del Producto tiene la autoridad para cancelar el Sprint, aunque puede hacerlo bajo la influencia de los interesados, del Equipo, o del ScrumMaster. ¿Bajo qué tipo de circunstancias puede un Sprint ser cancelado? La gerencia puede necesitar cancelar un Sprint si el objetivo del Sprint se queda obsoleto. Esto podría ocurrir si la empresa cambia de dirección, o si cambia el mercado o las condiciones de la tecnología. En general, un Sprint debe ser cancelado si dadas las circunstancias ya no tiene sentido. Sin embargo, debido a la corta duración de los Sprints, rara vez tiene sentido hacer esta cancelación.

Cuando un Sprint se cancela, cualquier elemento del Product Backlog que haya sido completado y "hecho", es revisado. Estos elementos son aceptados si representan un incremento potencialmente entregable. Si no es así, el elemento se vuelve a colocar en el Product Backlog con sus estimaciones iniciales. Cualquier trabajo realizado en él se asume como perdido. La cancelación de un Sprint consume recursos, ya que todo el mundo tiene que reagruparse en otra Reunión de Planificación de Sprint para iniciar otro Sprint. Las cancelaciones de Sprints son a menudo traumáticas para el equipo, y muy poco frecuentes.

Reunión de Planificación del Sprint

Durante Reunión de Planificación del Sprint la iteración es planificada. La reunión se restringe a un bloque de tiempo de ocho horas para un Sprint de un mes. Para Sprints más cortos, se debería reservar para esta reunión un tiempo proporcionalmente menor, aproximadamente el 5% de la longitud total del Sprint (por ejemplo, para un Sprint de dos semanas sería una Reunión de Planificación de cuatro horas). La Reunión de Planificación del Sprint y consta de dos partes. La primera parte, un bloque de cuatro horas, es cuando se decide lo que se hará durante el Sprint. La segunda parte (otro bloque de cuatro horas para un Sprint de un mes), es cuando el equipo determina cómo se va a convertir esta funcionalidad en un incremento del producto durante el Sprint.

Hay dos partes en la Reunión de Planificación del Sprint: la parte del "¿Qué?" y la parte del "¿Cómo?". Algunos Equipos Scrum combinan las dos. En la primera parte, el Equipo Scrum aborda la cuestión del "¿Qué?". El Propietario del Producto presenta al equipo la parte más prioritaria del Product Backlog. Trabajan juntos para determinar qué funciones se van a desarrollar durante el próximo Sprint. La información de entrada para esta reunión es el Product Backlog, el último incremento del producto, la capacidad del Equipo y el rendimiento anterior del Equipo. La cantidad de Backlog que el Equipo selecciona es una decisión del Equipo. Sólo el Equipo puede evaluar lo que puede lograr en el próximo Sprint.

Una vez seleccionado el Product Backlog, se define un Objetivo para el Sprint. El Objetivo del Sprint es una meta que se alcanzará mediante la implementación del Product Backlog. Es una declaración que sirve de orientación al equipo acerca de por qué se está construyendo el incremento. El Objetivo del Sprint es un subconjunto del objetivo de la entrega.

La razón para tener un Objetivo del Sprint es dar al equipo un margen de maniobra respecto a la funcionalidad. Por ejemplo, el Objetivo para un Sprint podría ser: "Automatizar la funcionalidad de modificación de la cuenta del cliente mediante de un middleware de transacciones seguras y recuperables." A medida que el Equipo trabaja, mantiene este objetivo en mente. A fin de satisfacer el objetivo, implementa la funcionalidad y la tecnología. Si el trabajo resulta ser más duro de lo que el equipo había previsto, entonces, el equipo colabora con el Propietario del Producto y sólo implementa parcialmente la funcionalidad.

En la segunda parte de la Reunión de Planificación del Sprint, el equipo aborda la cuestión del "¿Cómo?". Durante la segunda mitad de la Reunión de Planificación del Sprint

(bloque de cuatro horas para un Sprint de un mes), el Equipo debe determinar cómo convertirá en un incremento, el Product Backlog (el "Qué") seleccionado durante la reunión de planificación de Sprint. El Equipo por lo general comienza por el diseño del trabajo. Mientras diseñan, identifican tareas. Estas tareas son las porciones detalladas del trabajo necesario para convertir el Product Backlog en software que funciona. Las Tareas se deben descomponer para que se puedan completar en menos de un día. Esta lista de tareas se llama Sprint Backlog. El Equipo se organiza para asignar y realizar el trabajo

Consejo

Por lo general, sólo un 60-70% del total del Sprint Backlog será concebido en la reunión de planificación del Sprint. Para el resto se entrará en detalle más adelante, o se darán estimaciones de alto nivel que se descompondrán más adelante durante el Sprint.

contenido en el Sprint Backlog, ya sea durante la Reunión de Planificación del Sprint, o sobre la marcha durante el Sprint.

El Propietario del Producto está presente en la segunda parte de la Reunión de Planificación del Sprint para aclarar el Product Backlog y ayudar a llegar a posibles acuerdos. Si el Equipo determina que tiene demasiado trabajo, o demasiado poco, puede renegociar el Product Backlog con el Propietario del Producto. El Equipo también puede invitar a otras personas a estar presentes, con el fin de proporcionar asesoramiento técnico o de dominio. En esta reunión, un Equipo nuevo a menudo se da cuenta de que, o bien se hundirá, o saldrá a flote como un equipo, no individualmente. El Equipo se da cuenta de que debe apoyarse en sí mismo. Al aceptar esto, comienza a auto-organizarse para llegar a tener las características y el comportamiento de un verdadero equipo.

Revisión del Sprint

Al final del Sprint, se lleva a cabo una reunión de Revisión de Sprint. Esta es una reunión restringida a un bloque de tiempo de cuatro horas para un Sprint de un mes. Para Sprints de menor duración, hay que asignar proporcionalmente menos tiempo de la longitud total para esta reunión (por ejemplo, para dos semanas, la Revisión del Sprint sería de dos horas); esta reunión no debe consumir más de 5% del total del Sprint. Durante la Revisión de Sprint, el Equipo Scrum y las partes interesadas debaten sobre lo que se acaba de hacer. En base a eso, y a los cambios en el Product Backlog que se hayan hecho durante el Sprint, colaboran para determinar las próximas cosas que se podrían hacer. Se trata de una reunión informal, en la que la presentación de la funcionalidad está destinada a fomentar la colaboración para determinar qué hacer a continuación.

La reunión incluye por lo menos lo siguiente: El Propietario del Producto identifica lo que se ha hecho y lo que no se ha hecho. El Equipo analiza lo que salió bien durante el Sprint y cuáles son los problemas que encontró, y cómo resolvió estos problemas. El Equipo entonces muestra el trabajo que ha sido completado y responde

preguntas. El Propietario del Producto a continuación, analiza el Product Backlog en su estado actual. Proyecta las fechas probables de finalización con distintos supuestos de velocidad. Todo el grupo colabora entonces sobre lo que ha visto y lo que esto significa en cuanto a qué hacer a continuación. La Revisión de Sprint ofrece una valiosa aportación a la siguiente Reunión de Planificación de Sprint.

Retrospectiva del Sprint

Después de la Revisión del Sprint, y antes de la próxima Reunión de Planificación de Sprint, el Equipo Scrum mantiene una reunión Retrospectiva del Sprint. Es una reunión restringida a un bloque de tiempo de tres horas para Sprints de un mes (asignar tiempo proporcionalmente menor para Sprints de longitud menor). En esta reunión, el ScrumMaster alienta al Equipo Scrum a revisar, en el marco de proceso y prácticas de Scrum, su proceso de desarrollo, para que sea más eficaz y agradable para el próximo Sprint. Muchos libros documentan técnicas que son útiles para su uso en las Retrospectivas.

El propósito de la Retrospectiva es inspeccionar cómo fue el último Sprint en lo que respecta a las personas, relaciones, procesos y herramientas. La inspección debería identificar y priorizar los principales elementos que hayan ido bien y aquellos elementos que si se hiciesen de forma diferente, podrían producir mejoras. Entre estos elementos se incluyen la composición del equipo Scrum, la organización de las reuniones, las herramientas, la definición de "hecho", los métodos de comunicación, y los procesos para convertir los elementos del Product Backlog en algo "hecho". Al final de la Retrospectiva del Sprint, el Equipo Scrum debería haber identificado acciones concretas de mejora que se implementarán en el próximo Sprint. Estos cambios se convierten en la adaptación derivada de la inspección empírica.

Scrum Diario

Cada Equipo se reúne todos los días 15 minutos en una reunión de inspección y adaptación llamada Scrum Diario. El Scrum Diario se lleva a cabo a la misma hora y en el mismo lugar a lo largo de todos los Sprints. Durante la reunión, cada miembro del equipo, explica:

1. Lo que ha conseguido hacer desde la última reunión;
2. Lo que va a hacer hasta la próxima reunión, y
3. Qué obstáculos tiene en su camino.

Los Scrums Diarios mejoran las comunicaciones, eliminan otras reuniones, identifican y eliminan los impedimentos al desarrollo, destacan y promueven la rápida toma de decisiones y mejoran el nivel de conocimiento de los proyectos.

El ScrumMaster se asegura de que el Equipo mantiene la reunión. El Equipo es responsable de conducir el Scrum Diario. El ScrumMaster enseña al Equipo a mantener el Scrum Diario breve, mediante la aplicación de las reglas, y asegurándose de que la gente es breve. El ScrumMaster también hace cumplir la regla de que a las gallinas no se les permite hablar ni de ninguna manera interferir en el Scrum Diario.

El Scrum Diario no es una reunión de seguimiento del estado del proyecto. No es para cualquiera, sino sólo para la gente que trabaja en transformar los elementos del Product Backlog en un incremento (el Equipo). El Equipo se ha comprometido a un Objetivo del Sprint, y a los correspondientes elementos del Product Backlog. El Scrum Diario (con las tres preguntas) constituye una inspección de los progresos hacia ese Objetivo del Sprint. Por lo general se mantienen reuniones subsiguientes, para adecuar el trabajo futuro en el Sprint. La intención es optimizar la probabilidad de que el equipo logre su Objetivo. Ésta es una reunión clave de revisión y adaptación en el proceso empírico de Scrum.

Artefactos de Scrum

Los Artefactos de Scrum incluyen: el Product Backlog, el Burndown de entrega, el Sprint Backlog, y el Sprint Burndown.

Product Backlog y Burndown de Entrega

Los requisitos para el producto, que el/los Equipo/s está/n elaborando, están listados en el Product Backlog. El Propietario del Producto es responsable del Product Backlog, de su contenido, disponibilidad y priorización. El Product Backlog nunca está completo. La primera versión para el desarrollo, tan sólo establece los requisitos inicialmente conocidos, y que son entendidos mejor. El Product Backlog evoluciona a medida que el producto y el entorno en el que se utilizará evoluciona. El Product Backlog es dinámico, ya que cambia constantemente para identificar qué necesita el producto para ser adecuado, competitivo y útil. Mientras existe un producto, el Product Backlog también existe.

El Product Backlog representa todo lo necesario para desarrollar y lanzar un producto exitoso. Se trata de una lista de todas las características, funciones, tecnologías, mejoras y correcciones de errores que constituyen los cambios que se

harán al producto para futuras versiones. Los elementos del Product Backlog deben tener los siguientes atributos: una descripción, una prioridad, y una estimación. La prioridad está guiada por el riesgo, el valor y la necesidad. Hay muchas técnicas para evaluar estos atributos.

El Product Backlog está ordenado por prioridad. La parte más prioritaria del Product Backlog determina las actividades de desarrollo que se llevarán a cabo de forma inmediata. Cuanto mayor sea la

Consejo

Los elementos del Product Backlog frecuentemente se detallan como Historias de Usuario. Los Casos de Uso también son apropiados, pero son más apropiados para su uso en el desarrollo de software en entornos de vida-crítica o de misión-crítica.

prioridad, el elemento es más urgente, es sobre el que más se ha pensado, y es el que atesora mayor consenso en cuanto a su valor. La parte más prioritaria del Product Backlog está más clara, y tiene información más detallada que el Product Backlog de menor prioridad. Recibe mejores estimaciones, basadas en la mayor claridad y el mayor detalle. Cuanto menor sea la prioridad, menor es el detalle, hasta el punto de que puede hacerse difícil distinguir claramente un elemento de baja prioridad.

A medida que se utiliza un producto, que aumenta su valor,

y que el mercado proporciona retroalimentación, el Product Backlog se va revelando como una lista más amplia y exhaustiva. Los requisitos nunca dejan de cambiar. El Product Backlog es un documento vivo. Los cambios en los requisitos de negocio, las condiciones del mercado, la tecnología, y el personal, causan cambios en el Product Backlog. Para minimizar

la repetición del trabajo, sólo se deben detallar los elementos de más alta prioridad. Los elementos del Product Backlog que ocuparán a los Equipos en los próximos Sprints tienen un alto nivel de detalle,

Consejo

Los Equipos Scrum suelen pasar el 10% de cada Sprint preparando el Product Backlog para satisfacer la definición anterior del mismo. Cuando los elementos del Product Backlog han sido preparados con este nivel de granularidad, los que están en la parte superior del mismo (los de mayor prioridad, y mayor valor) se descomponen para que quepan en un Sprint. Han sido analizados y se ha reflexionado exhaustivamente acerca de ellos durante el proceso de preparación. Cuando se lleva a cabo la Reunión de Planificación de Sprint, estos elementos de máxima prioridad del Product Backlog son bien entendidos y se pueden seleccionar fácilmente.

Consejo

Las pruebas de aceptación se utilizan a menudo como un atributo más del Product Backlog. A menudo pueden sustituir a descripciones de texto más detalladas, que contengan una descripción comprobable, de lo que el elemento del Product Backlog debe hacer cuando esté completado.

habiendo sido descompuestos para que cualquier elemento pueda completarse dentro de la duración del Sprint.

A menudo, varios equipos Scrum trabajan juntos en el mismo producto. En ese caso, se utiliza un solo Product Backlog para describir el trabajo futuro en el producto. Se utiliza entonces un atributo del Product Backlog que agrupa a los elementos. La agrupación puede hacerse por conjunto de características, por tecnología o por arquitectura, y es utilizada a menudo por el Equipo Scrum como una forma de organizar el trabajo.

El gráfico de Burndown de la Entrega registra la suma del esfuerzo restante estimado del Product Backlog a lo largo del tiempo. El esfuerzo se estima en cualquier unidad de trabajo que el Equipo Scrum, y la organización, hayan decidido. La unidad de tiempo que se utiliza generalmente es el Sprint.

Durante la Planificación de la Entrega se calculan inicialmente las estimaciones de los elementos del Product Backlog y, también posteriormente, a medida que los elementos son creados. Durante la preparación del Product Backlog las estimaciones son examinadas y revisadas. Sin embargo, pueden ser actualizadas en cualquier momento. El equipo es responsable de todas las estimaciones.

Consejo

En algunas organizaciones, se va añadiendo más trabajo al Backlog del que se va completando. Esto puede crear una línea de tendencia que es plana o incluso inclinada hacia arriba. Para compensar este comportamiento y mantener la transparencia, se puede crear una nueva línea base al añadir o quitar trabajo. *Esta línea base debería incluir o excluir sólo cambios significativos, y debería estar bien documentada.*

Consejo

La línea de tendencia puede ser poco fiable durante los primeros dos o tres Sprints de una entrega, a menos que los Equipos hayan trabajado juntos antes, conozcan bien el producto, y entiendan la tecnología subyacente.

El Propietario del Producto puede influir en el Equipo, ayudando a comprender y optar por soluciones de compromiso, pero la estimación final la hace el Equipo. El Propietario del Producto mantiene publicados el Product Backlog y la gráfica de Burndown de Entrega actualizados en todo momento. Se puede trazar una línea de tendencia en la gráfica, basándose en el cambio en el trabajo restante.

Sprint Backlog y Sprint Burndown

El Sprint Backlog se compone de las tareas que el Equipo realiza para convertir los elementos del Product Backlog en un incremento "hecho". Muchas de ellas se desarrollan durante la Reunión de Planificación del Sprint. Constituyen todo el trabajo que el equipo identifica como necesario para cumplir con el Objetivo del Sprint. Los elementos del Sprint Backlog deben descomponerse. La descomposición debe ser suficiente para que los cambios en curso puedan ser entendidos en el Scrum Diario. El tamaño normal para un elemento del Sprint Backlog en el que se está trabajando es de un día o menos.

El equipo modifica el Sprint Backlog a lo largo de todo el Sprint, así como la parte de Sprint Backlog adicional que surja durante el Sprint. A medida que se van concretando tareas individuales, se puede descubrir que se necesitan más o menos tareas, o que una determinada tarea llevará más o menos tiempo de lo previsto. A medida que es necesario nuevo trabajo, el Equipo lo añade al Sprint Backlog. A medida que se va trabajando o se terminan tareas, se actualizan las horas de trabajo estimado restante para cada tarea. Cuando alguna tarea se revela como innecesaria, es eliminada. Sólo el Equipo puede cambiar su Sprint Backlog durante un Sprint. Sólo el Equipo puede cambiar el contenido, o las estimaciones. El Sprint Backlog es una instantánea muy visible y en tiempo real del

Consejo

Siempre que sea posible, dibuje a mano el gráfico de burndown en una hoja grande de papel expuesta en el área de trabajo del Equipo. Es más probable que los Equipos vean un gráfico grande y visible, a que vayan a consultarlo si está en Excel o en otra herramienta.

trabajo que el equipo tiene previsto realizar durante el Sprint, y pertenece exclusivamente al Equipo.

Hecho

Scrum requiere que los Equipos construyan un incremento de la funcionalidad del producto en cada Sprint. Este incremento debe ser potencialmente entregable, ya que el Propietario del Producto puede decidir liberar la funcionalidad de forma inmediata. Para ello, el incremento debe constituir una porción completa del producto. Debe estar "hecho". Cada incremento debería ser aditivo a todos los incrementos anteriores, y debería ser probado exhaustivamente, de modo que se asegure que todos los incrementos funcionan en conjunto.

Durante el desarrollo del producto, asegurar que una determinada funcionalidad está completada, podría llevar a alguien a asumir que ha sido al menos codificada limpiamente, refactorizada, sometida a pruebas unitarias, construida y sometida a pruebas de aceptación. Alguien distinto podría asumir tan solo que el código ha sido construido. Si alguien no conoce cuál es la definición de "hecho", los otros dos pilares del control empírico de procesos no funcionan. Cuando alguien describe algo como *hecho*, todo el mundo debe entender qué significa *hecho*.

Hecho define lo que el equipo quiere decir, cuando se compromete a "hacer" un elemento del Product Backlog en un Sprint. Algunos productos no contienen documentación, por lo que la definición de "hecho" no incluye a la documentación. Un incremento totalmente "hecho" incluye todo el análisis, diseño, refactorización, programación, documentación y pruebas para el incremento, y para todos los elementos del Product

Consejo

El trabajo "sin hacer" a menudo se acumula en un elemento del Product Backlog llamado "Trabajo sin hacer" o "Trabajo de Implementación." A medida que este trabajo se tiene en cuenta, el burndown del Product Backlog se mantiene más preciso que si no fuese acumulado.

Backlog correspondientes al incremento. Las pruebas incluyen testeo unitario, de sistema, de usuario y de regresión, así como pruebas no funcionales como pruebas de rendimiento, de estabilidad, de seguridad y de integración. Hecho incluye cualquier internacionalización. Algunos Equipos no son capaces en un principio de incluir todo lo requerido para la implementación en su definición de hecho. Esto debe quedar muy claro para el Propietario del Producto. Este trabajo restante deberá ser hecho antes de que el producto pueda ser implementado y utilizado.

REFLEXIONES FINALES

Algunas organizaciones son incapaces de construir un incremento completo en el plazo de un Sprint. Posiblemente no dispongan de la infraestructura de testeo automatizado para completar todas las pruebas. En este caso, se crean dos categorías para cada incremento: el trabajo "hecho" y el trabajo "sin hacer". El trabajo "sin hacer" es la parte de cada incremento que deberá ser completada posteriormente. El Propietario del Producto conoce exactamente qué está inspeccionando al final del Sprint, porque el incremento cumple la definición de "hecho", y el Propietario del Producto entiende esta definición. El trabajo "sin hacer" es añadido a un elemento del Product Backlog llamado "trabajo sin hacer", de modo que se acumula y se refleja correctamente en la gráfica de Burndown de entrega. Esta técnica promueve la transparencia durante el progreso hacia una entrega. La inspección y adaptación llevada a cabo en la Revisión del Sprint, serán tan precisas como lo sea esta transparencia.

Por ejemplo, si un Equipo no es capaz de realizar pruebas de rendimiento, regresión, estabilidad, seguridad, e integración, para cada elemento del Product Backlog, hay que calcular la proporción de este trabajo con respecto al trabajo que sí puede ser realizado (análisis, diseño, refacción, programación, documentación, pruebas unitarias y pruebas de usuario). Digamos que la proporción es seis partes de "hecho" y cuatro partes de "sin hacer". Si el Equipo termina

un elemento del Product Backlog de seis unidades de trabajo (teniendo en cuenta que el Equipo estima basándose en lo que conoce acerca de cómo llegar a “hecho”), cuatro unidades son añadidas al elemento del Product Backlog de “trabajo sin hacer” una vez que hayan terminado.

En cada Sprint, el trabajo “sin hacer” de cada incremento se va acumulando, y debe ser afrontado antes de liberar el producto. Este trabajo se acumula de forma lineal, aunque en realidad presenta algo de acumulación exponencial que depende de las características de cada organización. Al final de cada entrega, se añaden Sprints de lanzamiento con el fin de terminar este trabajo “sin hacer”. El número de estos Sprints no es predecible, en la medida que la acumulación de trabajo “sin hacer” no sea lineal.