

Marcos de Trabajo Ágil

SCRUM & KANBAN



Objetivo: Brindar información específica sobre cómo funciona el marco de trabajo ágil, sus elementos constitutivos y cada uno de los componentes que permiten llevar adelante un proyecto bajo estos parámetros.

Índice

Índice

1. Preliminares	3
1.1. Manifiesto Ágil	3
1.2. Interesados	3
1.3. Dominio de las Metodologías Ágiles	4
2. Scrum	4
3. Valores de Scrum	5
4. Pilares de Scrum	5
5. Equipo Scrum	6
5.1. Developers	6
5.2. Product Owner	7
5.3. Scrum Master	7
6. Sprint	8
7. Eventos de Scrum	9
7.1. Sprint Planning	9
7.2. Daily Scrum	10
7.3. Sprint Review	11
7.4. Sprint Retrospective	11
8. Artefactos de Scrum	11
8.1. Product Backlog	12
8.2. Sprint Backlog	13
8.3. Incremento	13
9. ¿Qué significa ser Ágil?	14
9.1. Ejemplos concretos	
10. Técnicas Ágiles	16
10.1 Kanban (Definición - Valores - Principios)	
10.2 Prácticas Generales de Kanban	

Introducción

Scrum es un marco de trabajo cuyo propósito es desarrollar, entregar y mantener productos complejos. Ayuda a las personas, equipos y organizaciones a generar valor a través de soluciones adaptativas para problemas complejos.

1. Preliminares

1.1 Manifiesto Ágil

En febrero de 2001, 17 críticos de los modelos de mejora del desarrollo de software basados en procesos, se reunieron en Snowbird, Utah, para tratar sobre técnicas y procesos para desarrollar software. En la reunión se utilizó el término *Métodos Ágiles* para definir a los métodos emergentes como alternativa a las metodologías tradicionales, que eran consideradas pesadas y rígidas por su carácter normativo y la fuerte dependencia de planificaciones detalladas previas al desarrollo.

Los integrantes de la reunión resumieron los principios sobre los que se basan los métodos alternativos en cuatro postulados, lo que ha quedado denominado como Manifiesto Ágil [1]. En él se expone:

Estamos descubriendo formas mejores de desarrollar software tanto por nuestra propia experiencia como ayudando a terceros. A través de este trabajo hemos aprendido a valorar:

- *Individuos e interacciones, sobre procesos y herramientas.*
- *Software funcionando, sobre documentación extensiva.*
- *Colaboración con el cliente, sobre negociación contractual.*
- *Respuesta ante el cambio, sobre seguir un plan.*

1.2 Interesados

Es el rol que representa a quienes no forman parte del Equipo Scrum y que son consumidores del producto en construcción o que pueden influir en él. Son quienes tienen interés en la realización del proyecto o producto; ayudan a descubrir, desarrollar, lanzar y promover el producto, presentando sus ideas sobre él y brindan una perspectiva del negocio y del entorno en el cual será utilizado.

El éxito no depende directamente del rol pero su participación si puede ser decisiva. Entre sus responsabilidades suelen cubrir el dar retroalimentación, claridad sobre el valor de negocio, asesorar, sugerir, colaborar y apoyar al proyecto.

El proceso de llevar las ideas de los interesados a requerimientos es responsabilidad del Product Owner, quién los debe plasmar como elementos a ser desarrollados luego de un proceso de refinamiento. Estas ideas deben ser comprendidas y definidas para luego plantear posibles soluciones y crear prototipos de dichas soluciones para finalmente transcribirlas como elementos para ser desarrollados.

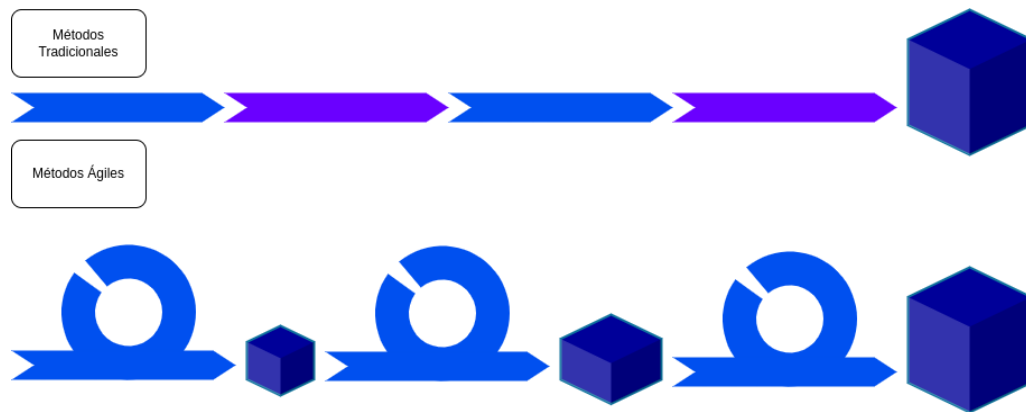


Figura 1: Métodos Tradicionales y Ágiles

1.3 Dominio de las Metodologías Ágiles

Debido a las características de las metodologías ágiles, estas son apropiadas para cualquier tipo de proyecto con requisitos inestables y para los que requieren rapidez y flexibilidad. Por consiguiente, hoy en día son aplicadas e implementadas en diferentes áreas además del desarrollo de productos de software.

2. Scrum

Es un framework o marco de trabajo desarrollado bajo el manifiesto ágil, que ayuda a las personas, equipos y organizaciones a generar valor a través de soluciones adaptativas para problemas complejos.

Este modelo fue identificado y definido por Ikujiro Nonaka y Takeuchi a principios de los 80, al analizar cómo desarrollaban los nuevos productos las principales empresas de manufactura tecnológica [2].

Si bien comenzó a hablarse de él en 1986 no fue hasta 1995 que fue propuesto formalmente por Ken Schwaber y Jeff Sutherland, enfocado en equipos autogestionados y multifuncionales con entregas iterativas y funcionales del producto, basado en el aprendizaje empírico, sin reglas definidas o pautas a seguir para conseguir el resultado esperado, sino que está en desarrollo continuo..

Es un marco de trabajo ligero, fácil de entender pero difícil de dominar, dentro del cual se pueden emplear varios procesos técnicos y consiste de un conjunto de reglas, eventos, artefactos y un Equipo Scrum. Se basa en la inteligencia colectiva de las personas que lo utilizan. En lugar de proporcionar a las personas instrucciones detalladas, las reglas de Scrum guían sus relaciones e interacciones.

Teoría de Scrum: Scrum se basa en la teoría de control de procesos empíricos o empirismo y el pensamiento Lean. El empirismo asegura que el conocimiento proceda de la experiencia y de tomar decisiones basándose en evidencias. Mientras que el pensamiento Lean reduce el desperdicio y se enfoca en lo esencial.

Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo. Involucra a grupos de personas que colectivamente tienen todas las habilidades y experiencia para hacer el trabajo y compartir o adquirir dichas habilidades según sea necesario

3. Valores de Scrum

El uso exitoso de Scrum depende de que las personas se vuelvan más competentes en vivir los siguientes valores:

Compromiso, sentido de responsabilidad con la entrega continua de valor y las metas.

Coraje, para cumplir con los procesos y compromisos incluso en momentos difíciles.

Foco, concentración para conseguir las metas y objetivos propuestos.

Apertura, a nuevas ideas y perspectiva de las cosas.

Respeto, entre los miembros del equipo, a los compromisos y procesos.

Estos valores dan dirección al equipo con respecto a su trabajo, acciones y comportamiento.

4. Pilares de Scrum

Los pilares establecen los parámetros para la experimentación y validación de los hechos concretos registrados en los diferentes artefactos de Scrum a lo largo del proceso. Son estrategias que se pueden implementar para obtener los mejores resultados.

Transparencia

La transparencia facilita la toma de decisiones, es por ello que todos los artefactos y eventos de Scrum deben ser visibles para todo el Equipo Scrum y es responsabilidad del equipo velar por esto.

Todos los miembros del Equipo Scrum deben saber qué se está haciendo, el estado del proyecto y los resultados que se están obteniendo, es decir, visibilidad completa del proyecto y de su salud, y debe estar definido por un estándar, de forma a tener un entendimiento único.

Inspección

Se debe inspeccionar frecuentemente los artefactos de Scrum y el progreso con el fin de detectar variaciones indeseadas. Es la capacidad de realizar observaciones, mediciones y análisis para saber el estado del proyecto en sus diferentes aspectos y de ajustar algún proceso si fuese necesario.

Adaptación

Conforme se avanza en el ciclo de desarrollo se pueden realizar cambios o ajustes que se crean necesarios. Si se determina que un proceso o algún aspecto se desvía de los límites aceptables, este debe ajustarse lo antes posible con el fin de minimizar desviaciones mayores.

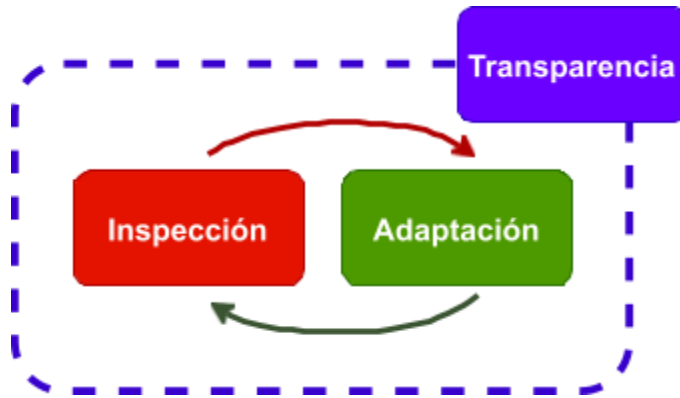


Figura 2: Pilares de Scrum

Estos pilares se complementan y se relacionan entre sí para mejorar continuamente los procesos Scrum. La inspección permite la adaptación y la adaptación debe ser inspeccionada, formando un flujo continuo de mejora, y a su vez, todo este proceso debe ser transparente para todo el Equipo Scrum.

5. Equipo Scrum

Son los responsables de entregar el producto de forma iterativa e incremental, un incremento valioso y útil en cada Sprint. El Equipo Scrum debe ser autogestionado y multifuncional, debe estar diseñado para optimizar la flexibilidad, creatividad y la productividad, aprovechando los escenarios de retroalimentación para mejorar los procesos.

El equipo debe ser lo suficientemente pequeño como para seguir siendo ágil y lo suficientemente grande como para completar un trabajo significativo dentro de un Sprint, generalmente 10 personas o menos.

Scrum define tres responsabilidades específicas dentro del equipo: los Developers, el Product Owner y el Scrum Master.

5.1 Developers

Es el equipo encargado de desarrollar los elementos que se encuentran dentro del sprint backlog, que es la lista de elementos que fueron seleccionados para ser desarrollados en un Sprint. El equipo debe ser autogestionado y multifuncional, a su vez que debe gozar de

cierta autonomía. Todos los miembros deberían ser capaces de realizar todas las tareas de desarrollo.

Son responsables de entregar un incremento terminado del producto, listo para ser presentado a los interesados al final de cada Sprint. Son los únicos responsables del incremento y nadie puede forzarlos a trabajar en un conjunto diferente de requisitos.

Al hablar de un equipo es importante mencionar que Scrum no reconoce títulos dentro del equipo, ni

tampoco reconoce subequipos dentro del mismo.

5.2 Product Owner

Es responsable de maximizar el valor del producto resultante de los developers, debe tener un entendimiento claro de la organización y del negocio. Es una persona, no un comité y debe ser respetado por toda la organización.

Debe ayudar a los interesados a plasmar las ideas en requerimientos y construirlos como elementos del product backlog; lista de las características, funcionalidades y requerimientos que implican cambios a realizarse en el producto, para que sean desarrollados en el Sprint. Único responsable de gestionar el product backlog y capaz de cancelar un Sprint.

Entre sus principales funciones se encuentran:

- Expresar claramente los elementos del product backlog. Priorizar los elementos del product backlog.
- Consolidar la visibilidad y transparencia del product backlog.
- Asegurar que los developers entiendan claramente los elementos del product backlog.

5.3 Scrum Master

Facilitador que obra para que todos los eventos de Scrum se realicen correctamente y de forma eficiente, trabaja para promover y apoyar la metodología Scrum, ayudando a entender la teoría, prácticas, reglas y valores, y busca maximizar las interacciones entre el equipo.

Debe proteger al Sprint y los developers, es decir, el compromiso que el equipo ha tomado al comienzo del Sprint. No debería trabajar con los interesados de forma directa.

Al ser un facilitador se relaciona con los demás miembros del Equipo Scrum y con la organización misma. Sus principales funciones para con el **Product Owner**:

- Asegurar la transparencia y el entendimiento de los objetivos y alcance del producto.
- Encontrar técnicas para gestionar el product backlog.
- Planificar en un entorno empírico.

Sus principales funciones para con los **Developers**:

- Guiar al equipo para que este sea autogestionado y multifuncional.
- Ayudar al equipo a crear un producto de alto valor.
- Eliminar impedimentos para el progreso del equipo.

Sus principales funciones para con la **Organización**:

- Liderar y guiar la organización en la adaptación de Scrum. Implementar Scrum

en la organización.

- Ayudar a los interesados dentro de la organización a entender e implementar Scrum.

6. Sprint

Es el ciclo de iteración de Scrum, es el corazón de él, consiste en un *timebox* o bloque de tiempo de 1 a 4 semanas, este periodo se define acorde a la naturaleza del proyecto y se recomienda que todos los Sprints tengan la misma duración. Cada Sprint puede considerarse un proyecto corto.

Cada Sprint está constituido por su propio sprint backlog, que se define en la planificación o *Sprint Planning* entre el Product Owner y los developers. Cada elemento del Sprint debe ser independiente en lo posible, sin dudas por aclarar, sin ningún bloqueo o dependencia de otro elemento y listo para ser desarrollado, esto se conoce como *Definition of Ready*.

Durante la ejecución de un Sprint, también se realiza el refinamiento del product backlog, si fuese necesario, este trabajo se realiza en conjunto entre el Product Owner y los developers.

Al finalizar un Sprint debe existir un incremento del producto, un aumento de valor que debe ser validado por los interesados durante el *Sprint Review*. Cada nuevo Sprint comienza inmediatamente luego de la finalización del Sprint anterior.

El seguimiento del Sprint se realiza por medio del *Daily Scrum* con el fin de gestionar el progreso del mismo.

Durante el Sprint

- No se realizan cambios que puedan afectar al objetivo del Sprint.
- No se deben disminuir los objetivos de calidad.
- El product backlog se refina según sea necesario.
- El alcance puede clarificarse y renegociarse entre el Product Owner y los developers.

Cancelación del Sprint

Un Sprint puede cancelarse si el objetivo del Sprint ha quedado obsoleto. Solo puede ser cancelado por el Product Owner.

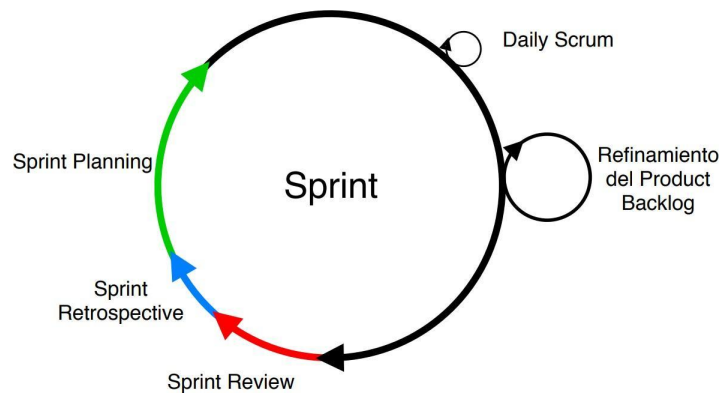


Figura 3: Eventos de Scrum realizados dentro del Sprint

7. Eventos de Scrum

Estos eventos existen con el fin de crear regularidad y evitar reuniones no definidas. Cada uno de los eventos debe verse como una oportunidad para la inspección y adaptación de los procesos.

Todos los eventos de Scrum trabajan con un *time box* o bloque de tiempo, es decir, una duración máxima para dicho evento. Si bien la duración del Sprint es fija, los demás eventos pueden darse por concluidos cuando estos alcanzan sus objetivos correspondientes.

Se reconocen como eventos oficiales de Scrum, el Sprint Planning, Daily Scrum, Sprint Review y Sprint Retrospective.

7.1 Sprint Planning

Reunión donde se planifica el trabajo que será realizado durante el Sprint por los developers, este plan se crea mediante el trabajo del Equipo Scrum.

El Product Owner se asegura de que cada participante esté preparado para discutir los elementos más importantes del Product Backlog y cómo se relacionan con el Objetivo del Producto. También puede invitar a otras personas a asistir al Sprint Planning para brindar asesoramiento. El Scrum Master debe velar por la realización de este evento y deben participar el Product Owner y los developers.

En general se recomienda un bloque de tiempo de 8 horas para un Sprint de 4 semanas, en caso que el Sprint tenga una duración menor, el Sprint Planning también debe tener una duración menor.

En este evento se aborda los siguientes temas:

¿POR QUÉ ES VALIOSO ESTE SPRINT?

El Product Owner propone cómo el producto podría incrementar su valor y utilidad en el Sprint actual. Luego, todo el equipo colabora para definir un objetivo del Sprint, que

la meta establecida para el Sprint y que se logra mediante el desarrollo de los elementos del sprint backlog. El Objetivo del Sprint debe completarse antes de que termine el Sprint Planning.

¿QUÉ SE PUEDE HACER EN ESTE SPRINT?

Para poder identificar el trabajo que puede realizarse en el Sprint, se consideran como datos de entrada de este evento los elementos del product backlog, la capacidad proyectada de los developers, el incremento y el rendimiento del Sprint anterior, con el fin de tener una visión más precisa de la capacidad del equipo, esto puede variar de un Sprint a otro.

Se deben priorizar los elementos del product backlog para lograr el cumplimiento del objetivo del sprint, este objetivo debe ser alcanzado por los developers por medio del desarrollo de los elementos del sprint backlog.

Solo los developers pueden evaluar su capacidad para lograr el objetivo durante el Sprint que comienza y cuáles elementos serán seleccionados para ser desarrollados.

¿CÓMO SE CONSEGUIRÁ COMPLETAR EL TRABAJO?

Los elementos seleccionados del product backlog para el Sprint, más el plan para terminarlos recibe el nombre de sprint backlog.

Durante el Sprint Planning se hace una proyección de lo que se cree que se puede completar en un Sprint. El trabajo puede estimarse con diferentes técnicas y queda a criterio de los developers seleccionar la más adecuada.

Es responsabilidad de los developers definir cómo construir los elementos del product backlog, el incremento del Sprint, mientras que el Product Owner es responsable de esclarecer las ideas y de negociar el sprint backlog con los developers.

7.2 Daily Scrum

Son reuniones diarias que se realizan cada 24 horas, o bien una vez al día con el fin de conocer la salud del Sprint, inspeccionando el trabajo realizado, los impedimentos y la proyección del mismo, evaluando el progreso hacia el cumplimiento del objetivo del Sprint, y adaptar el sprint backlog según sea necesario.

Es una reunión con una duración máxima de 15 minutos y se recomienda realizarlo al comienzo del día, pero también es posible consensuar un horario para realizarlo. Los developers son responsables de llevar a cabo esta reunión y de seleccionar la estructura y las técnicas que deseen, ya sea utilizando preguntas y/o discusiones, tales como:

¿Qué hice ayer?

¿Qué haré hoy?

¿Existen impedimentos?

Estas reuniones mejoran la comunicación, identifican impedimentos, promueven la toma rápida de decisiones y, en consecuencia, eliminan la necesidad de otras reuniones. Si hubiese necesidad después del Daily Scrum se deben desarrollar discusiones más detalladas sobre cómo adaptar o volver a planificar el resto del trabajo del Sprint.

7.3 Sprint Review

El propósito de esta reunión es inspeccionar el resultado del Sprint, el incremento del producto y adaptar el product backlog si fuese necesario para el siguiente Sprint. En esta reunión participan los interesados y el Equipo Scrum con el fin de facilitar la retroalimentación.

El Equipo Scrum presenta los resultados de su trabajo a los interesados y se discute el progreso hacia el Objetivo del Producto. El Product Owner realiza una breve introducción, un resumen del Sprint que finalizó y los elementos que fueron desarrollados, mientras que los developers son los responsables de presentar los elementos que fueron terminados durante la ejecución del Sprint para ser validados por los interesados. Deben presentar el incremento, realizar una demostración del mismo y explicar los detalles de implementación si son requeridos, así como los problemas presentes durante el Sprint.

En general se recomienda un bloque de tiempo de 4 horas para un Sprint de 4 semanas, en caso que el Sprint tenga una duración menor, el Sprint Review también debe tener una duración menor.

El Product Owner realiza un cierre mostrando el estado actual del product backlog, ilustrando las proyecciones de los siguientes Sprints y el estado actual del producto. El resultado de esta reunión es un product backlog revisado y priorizado.

7.4 Sprint Retrospective

Reunión para inspeccionar al equipo y planificar formas de aumentar la calidad y la efectividad, tiene lugar después del Sprint Review y antes del Sprint Planning del Sprint entrante.

En esta reunión se presentan todos los inconvenientes presentes durante la ejecución de un Sprint, con el fin de aprender de ellos e ir mejorando en cada iteración, se debe diseñar un plan de mejora y aplicarlo durante la ejecución del siguiente Sprint.

Se inspecciona cómo fue el último Sprint con respecto a las personas, las interacciones, los procesos, las herramientas y su Definición de Terminado. Se identifican los supuestos que los llevaron por mal camino y se exploran sus orígenes. El Scrum Team analiza qué salió bien durante el Sprint, qué problemas encontró y cómo se resolvieron (o no) esos problemas.

Con esta reunión concluye el Sprint. Se recomienda un bloque de tiempo de 3 horas para un Sprint de 4 semanas, en caso que el Sprint tenga una duración menor, el Sprint Review debe durar menos.

8. Artefactos de Scrum

Son elementos útiles para proporcionar transparencia y oportunidades para la inspección y adaptación del trabajo y el Equipo Scrum, brindan información sobre diferentes aspectos del producto durante su desarrollo.

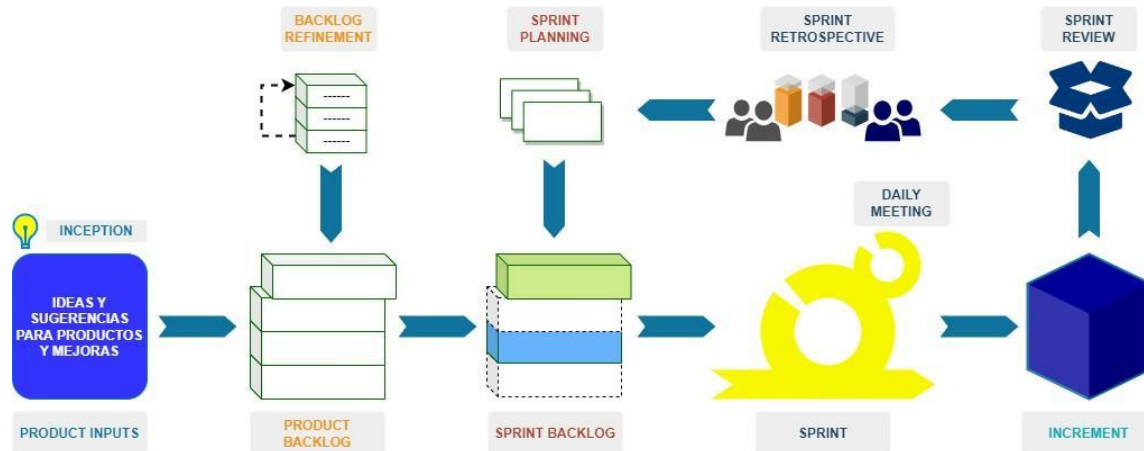


Figura 4: Eventos y Artefactos de Scrum

Cada uno contiene un compromiso para garantizar que proporcione información que mejore la transparencia y el enfoque frente al cual se pueda medir el progreso:

Para el **Product Backlog**, es el **Objetivo del Producto**. Para el

Sprint Backlog, es el **Objetivo del Sprint**.

Para el **Incremento** es la **Definición de Terminado**.

Estos compromisos existen para reforzar el empirismo y los valores de Scrum para el Scrum Team y sus interesados.

8.1 Product Backlog

Es la lista de elementos que enumera todas las características, funcionalidades, requerimientos, mejoras y correcciones que constituyen cambios a realizarse en el producto en entregas futuras. Es la única fuente para cualquier cambio a realizarse, el Product Owner es el responsable del contenido, disponibilidad y orden.

Evoluciona a medida que el producto y el entorno que lo usará lo hacen, es volátil acorde a las necesidades, ideas o requisitos del producto o del entorno. Cambios en los requisitos de negocio, condiciones de mercado o de tecnología podrían ocasionar cambios y/o ajustes en el product backlog.

Los elementos deben estar listos para formar parte del sprint backlog, deben tener cierto grado de detalle, transparencia e independencia. Cada elemento debería de tener, mínima- mente, descripción, orden, estimación y valor agregado, y si es posible demostraciones de la completitud del mismo.

Es posible que varios equipos trabajen en el mismo producto, pero de igual manera se debería de mantener un solo product backlog pero agrupando los elementos por algún atributo común entre ellos.

Los elementos deben ser descompuestos con el fin de que puedan ser desarrollados dentro de un Sprint, es decir, puedan ser terminados en el tiempo límite del Sprint. Generalmente los elementos con mayor prioridad son los que están mejor detallados, que facilita la estimación de los mismos.

El product backlog es desarrollado por los developers en Sprints, agrupando los elementos en sprint backlogs. Los developers son responsables de las estimaciones de dichos elementos.

OBJETIVO DEL PRODUCTO

Es el objetivo a largo plazo del Equipo Scrum. Describe un estado futuro del producto que sirve como objetivo para que el equipo planifique.

Un producto es un vehículo para entregar valor. Tiene un límite claro, personas interesadas conocidas, usuarios o clientes bien definidos. Un producto puede ser un servicio, un producto físico o más abstracto.

Refinamiento

Es el acto de añadir detalles, estimaciones y prioridad a los elementos del product backlog, se realiza en colaboración entre el Product Owner y los developers, y se recomienda que se utilice el 10 % de la capacidad de los developers, sin embargo el Product Owner puede actualizar los elementos en cualquier momento.

8.2 Sprint Backlog

Se compone del Objetivo del Sprint (el por qué), el conjunto de elementos del product backlog seleccionados para ser desarrollados en el Sprint (el qué), así como un plan de acción para entregar el incremento del producto (el cómo).

El sprint backlog es un plan realizado por y para los developers. Es una imagen visible y en tiempo real del trabajo que los developers planean realizar durante el Sprint para lograr el objetivo del Sprint. Solo los developers pueden cambiar los elementos del Sprint durante la ejecución del mismo.

OBJETIVO DEL SPRINT

Es el único propósito del Sprint. Si bien el Objetivo del Sprint es un compromiso de los developers, proporciona flexibilidad en términos del trabajo exacto necesario para lograrlo.

Es definido durante el Sprint Planning y se agrega al sprint backlog. Si el trabajo resulta ser diferente a lo esperado, los developers colaboran con el Product Owner para negociar el alcance del sprint backlog dentro del Sprint sin afectar el objetivo.

8.3 Incremento

Un incremento es el conjunto de todos los elementos del Sprint terminados y el valor de todos los incrementos de los Sprint anteriores. Debe estar listo para ser utilizado independiente de si el Product Owner desea ponerlo o no a disposición de los interesados.

Un elemento no puede considerarse parte de un incremento a menos que cumpla con la Definición de Terminado.

Definición de Terminado (DONE)

Son las características que debe cumplir un elemento para que pueda considerarse finalizado o terminado correctamente. Es una descripción formal para identificar que los elementos del Sprint fueron finalizados correctamente.

Crea transparencia al brindar a todos un entendimiento compartido de qué trabajo se completó como parte del incremento. Si un elemento no cumple con la Definición de Terminado, no se puede publicar ni presentar en el Sprint Review. En el momento en que un elemento del Product Backlog cumple con la Definición de Terminado, nace un incremento.

Si existen varios equipos, trabajando en un mismo producto o proyecto, estos deben definir en conjunto dicho concepto, los miembros del Equipo Scrum deben tener y deben manejar un solo concepto de Definición de Terminado. A medida que el Equipo Scrum madura, se espera que esta definición se amplíe para incluir criterios más sólidos y rigurosos.

9.0 ¿Qué significa SER ágil?

El aprendizaje de cualquier habilidad tiene tres etapas:

Shu: se elige una técnica y, asumiendo que es correcta, se intenta imitar.

Ha: se coleccionan más técnicas.

Ri: se experimenta e inventan nuevas técnicas mezclando, combinando y modificando.

Las técnicas de etapa shu son aplicables en general. Las técnicas de etapa ri sólo funcionan en casos concretos, y requieren de conocimiento experto para saber cuándo y cómo aplicarlas. «No puedes ganar en una industria competitiva utilizando técnicas shu.» Alistair Cockburn

10. Técnicas Ágiles

10.1 Kanban (Definición - Valores)

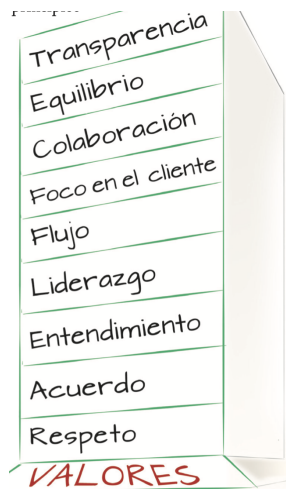
DEFINICIÓN

Kanban es un método para definir, gestionar y mejorar servicios que entregan trabajo del conocimiento, tales como servicios profesionales, trabajos o actividades en las que interviene la creatividad y el diseño tanto de productos de software como físicos. Se caracteriza por el principio de “empieza por donde estés” — por medio del cual se consigue catalizar el cambio rápido y focalizado dentro de las organizaciones — que reduce la resistencia a un cambio favorable en línea con los objetivos de la organización.

El método Kanban se basa en hacer visible lo que de otro modo es trabajo del conocimiento intangible, para asegurar que el servicio funciona con la cantidad de trabajo correcta — trabajo que es requerido y necesitado por el cliente y que el servicio tiene la capacidad de entregar. Para realizar esto, utilizamos un sistema kanban — un sistema de flujo de entrega que limita la cantidad de trabajo en progreso (WiP, del inglés Work In Progress) utilizando señales visuales.

El mecanismo de señalización, a veces referido como *kanbans*, se muestra en tableros kanban y representa los límites del trabajo en progreso, los cuales previenen cuanto de más o de menos trabajo entra en el sistema, de este modo mejora el flujo de valor a los clientes. Las políticas para limitar el WiP crean un sistema de arrastre: el trabajo es “arrastrado” al sistema cuando otro de los trabajos es completado y queda capacidad disponible, en lugar de “empujar” estos trabajos al sistema cuando hay nuevo trabajo demandado.

VALORES



El Método Kanban está guiado por valores. Está motivado por la creencia de que es necesario respetar a todos los individuos que contribuyen colaborativamente en una organización, no solo para el éxito de la empresa, sino también para que todo merezca la pena.

Los valores de Kanban se podrían resumir en una sola palabra, “respeto”. Sin embargo, es importante desglosar esto en una serie de nueve valores (incluyendo respeto) que encapsulan el porqué de la existencia de los principios.

Estos valores encarnan las motivaciones de Kanban en la búsqueda de la mejora de los servicios realizados por equipos colaborativos. El método no puede ser aplicado fielmente sin abrazarlos como un todo.

10.2 Prácticas Generales de Kanban

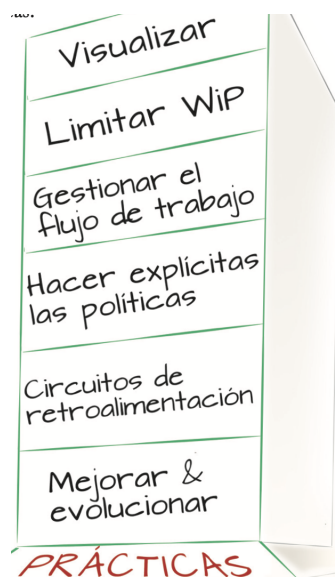
Las prácticas generales de Kanban definen las actividades fundamentales para el manejo de los sistemas kanban.

Estas seis son:

1. Visualizar.
2. Limitar el trabajo en progreso.
3. Gestionar el flujo.
4. Hacer explícitas las políticas.
5. Implementar ciclos de retroalimentación o feedback.
6. Mejorar de manera colaborativa, evolucionar experimentalmente.

Estas prácticas suponen:

- *ver* el trabajo y las políticas que determinan como este se procesa; entonces
- *mejorar* el proceso de una manera evolutiva — manteniendo y amplificando cambios útiles y aprendiendo incluso invirtiendo o disminuyendo cambios ineficaces.



Un tablero kanban es una manera, pero no la única, de visualizar el trabajo y el proceso por el que este discurre. Para que sea un sistema kanban, más que un simple sistema de flujo, los compromisos y las entregas tienen que estar definidos, así como los límites del trabajo en progreso en cada una de las etapas del tablero, desde el inicio del trabajo a la entrega. El hecho de hacer el trabajo y las políticas que lo gobiernan visibles — bien en un tablero en la pared, en un tablero virtual, o de otras maneras — es el resultado de haber recorrido un camino importante de colaboración para entender el sistema actual y encontrar áreas de mejora potenciales.

También es importante visualizar las políticas; por ejemplo, colocando resúmenes entre las columnas donde se describe lo que debe estar hecho antes de que una tarjeta se mueva de una columna a la siguiente.

El diseño de los tableros varía mucho entre un sistema kanban y otro, dependiendo de cómo se utilizan. El método no restringe cómo diseñar los tableros. Las herramientas de software desarrolladas para dar soporte a Kanban pueden introducir algunas restricciones prácticas — por ejemplo, el patrón común de una cuadrícula de dos dimensiones con tarjetas mostrando información sobre cada elemento de trabajo. Las columnas representan pasos en el proceso, y algunas de las columnas están divididas horizontalmente (en calles, si atraviesan una o más columnas) para distinguir estados de los elementos dentro de los pasos.

REFERENCIAS

- KANBAN ESENCIAL CONDENSADO - David Anderson & Andy Carmichel

Lean KANBAN, Press Seattle, WA. First digital version, 17 April, 2016. This version 28 July 2016.