

Estructuras, Metodologías y Métodos Ágiles y Lean

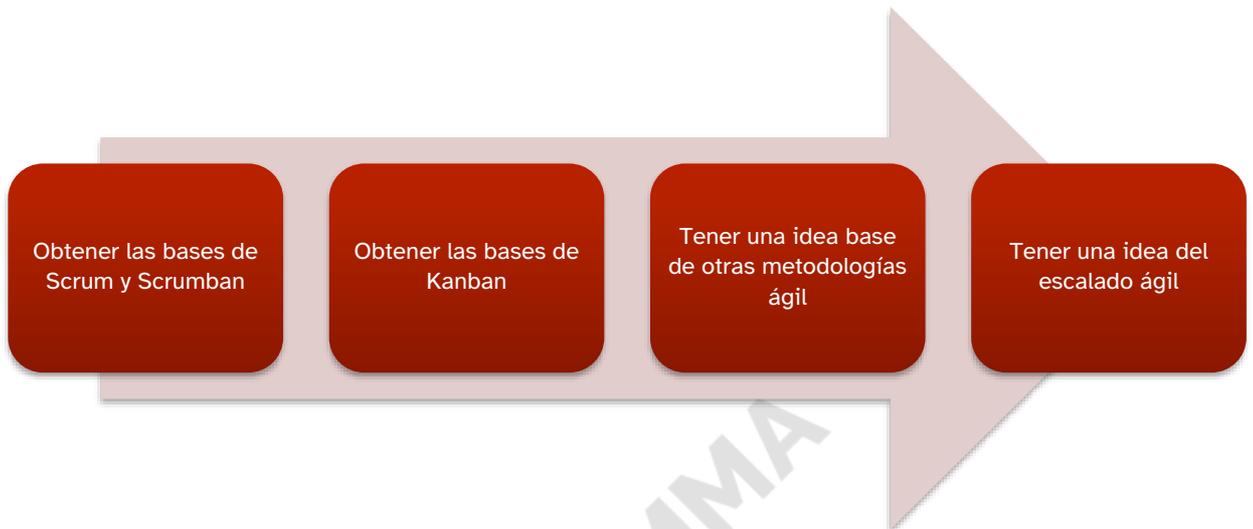
Índice

1	Scrum	4
1.1	La estructura de Scrum.....	4
1.2	Gestión con Scrum.....	8
2	Kanban.....	11
2.1	Las seis prácticas de Kanban.....	11
2.1.1	Alta visualización.....	11
2.1.2	Limitar la cantidad de trabajo en progreso.....	12
2.1.3	Mejorar colaboradoramente y evolucionar experimentalmente.....	13
2.1.4	Gestionar el flujo.....	14
2.1.5	Hacer los procesos explícitos.....	15
2.1.6	Implementar ciclos de retroalimentación.....	15
2.2	Gestión con Kanban	16
2.3	Kanban personal.....	18
3	Scrumban.....	19
4	Otras Metodologías	20
4.1	DSDM	20
4.2	Programación extrema	21
5	Escalado de Ágil	22
5.1	SAFe	22

5.2	LESS.....	23
5.3	Niveles de vuelo.	24
6	Conclusiones.....	25
7	Bibliografía.....	26

Red SUMMA ©

Objetivos



1 Scrum

1.1 La estructura de Scrum.

Como se vio en la Clase 1, ágil es una forma de pensar y de actuar. Scrum es una estructura que sirvió como una de las bases para elaborar el manifiesto ágil y la declaración de interdependencia. Scrum para desarrollo de software fue desarrollado principalmente por Jeff Sutherland y Ken Schwaber a inicios de los 1990's [Sutherland]. Su origen, sin embargo, viene del artículo de Hirotaka Takeuchi e Ikujiro Nonaka titulado "The new new product development game" (efectivamente con la palabra "new" dos veces) publicado en la revista Harvard Business Review en 1986 [Takeuchi & Nonaka].

La estructura de Scrum es:

Scrum es una estructura para la gestión de desarrollo de software que reemplaza la gestión existente en la organización.

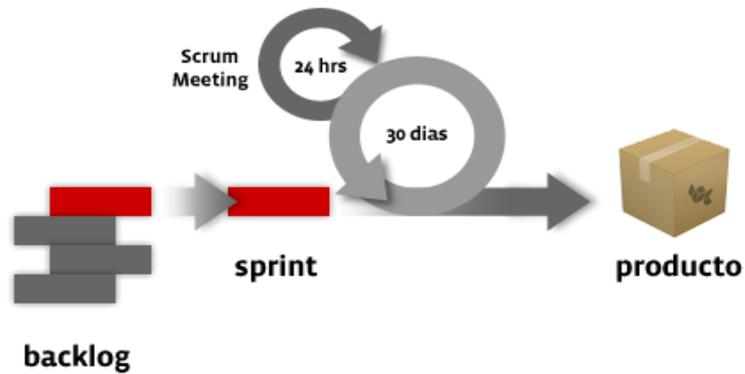
Ha sido utilizada en otros contextos.

Scrum tiene:

3 roles

5 eventos

2 artefactos



La estructura incluye también roles, eventos y artefactos:

Product Owner

- Responsable de maximizar el valor del producto.
- Gestión de la cartera de productos.
- Requisitos Definidos.
- Priorización.
- Comprensión compartida, valor y visibilidad.

Scrum Master

- Asegura que Scrum se utiliza correctamente.
- Líder de servicio.
- Eliminar los impedimentos.
- Facilitar eventos.
- Entrenar al equipo.
- Asiste al propietario del producto con el backlog de Producto.

Equipo de Desarrollo

- Genere los incrementos del producto.
- Autoorganización
- Multifuncional (reúne todas las competencias necesarias para el desarrollo del product).
- No más de 7-8 personas

ROLES
Equipo de desarrollo
Product Owner
Scrum Master

Planificación de la entrega.

- El alcance completo
- Backlog de producto priorizado
- Determinar la duración del sprint.

Planificación del Sprint

- Lo que entregamos en ese Sprint.
- El equipo de desarrollo pronostica lo que se puede entregar (estimaciones, capacidad, velocidad) Dos partes: Estratégico + Táctico.
- Al final, el equipo sabrá lo que tiene que construir y cómo.

Scrum

- Timeboxed: 15 min
- Mismo tiempo, mismo lugar.
- Equipo de desarrollo + SM
- ¿Qué se ha logrado desde la última reunión?
- ¿Qué se hará antes de la próxima reunión?
- ¿Qué obstáculos hay en el camino?

Revisión del Sprint

- Al final de Sprint.
- Inspecciona el incremento del producto.
- Demuestra el trabajo que se ha hecho.
- Todo el equipo.
- El Product Owner Rechaza o Aceptar.
- Determina qué hacer a continuación.

EVENTOS

Planificación de la entrega
Planificación del Sprint
Scrum Diario
Revisión del Sprint
Retrospectiva del Sprint

Retrospectiva del Sprint

- Después de la Revisión del Sprint.
- Todo el equipo participa en una reunión que busca la mejora.
- Lo que salió bien, lo bien mal, identificar las oportunidades de mejora para implementar en el próximo sprint.
- Timeboxed con un facilitador de la reunión.

	Item #	Description	Est	By
Very High				
	1	Finish database versioning	16	KH
	2	Get rid of unneeded shared Java in database	8	KH
		- Add licensing	-	-
	3	Concurrent user licensing	16	TG
	4	Demo / Eval licensing	16	TG
		Analysis Manager		
	5	File formats we support are out of date	160	TG
	6	Round-trip Analyses	250	MC
High				
		- Enforce unique names	-	-
	7	In main application	24	KH
	8	In import	24	AM
		- Admin Program	-	-
	9	Delete users	4	JM
		- Analysis Manager	-	-
		When items are removed from an analysis, they should show up again in the pick list in lower 1/2 of the analysis tab		
	10		8	TG
		- Query	-	-
	11	Support for wildcards when searching	16	T&A
	12	Sorting of number attributes to handle negative numbers	16	T&A
	13	Horizontal scrolling	12	T&A
		- Population Genetics	-	-
	14	Frequency Manager	400	T&M
	15	Query Tool	400	T&M
	16	Additional Editors (which ones)	240	T&M
	17	Study Variable Manager	240	T&M
	18	Haplotypes	320	T&M
	19	Add icons for v1.1 or 2.0	-	-
		- Pedigree Manager	-	-
	20	Validate Derived kindred	4	KH
Medium				
		- Explorer	-	-
		Launch tab synchronization (only show queries/analyses for logged in users)		
	21		8	T&A
	22	Delete settings (?)	4	T&A

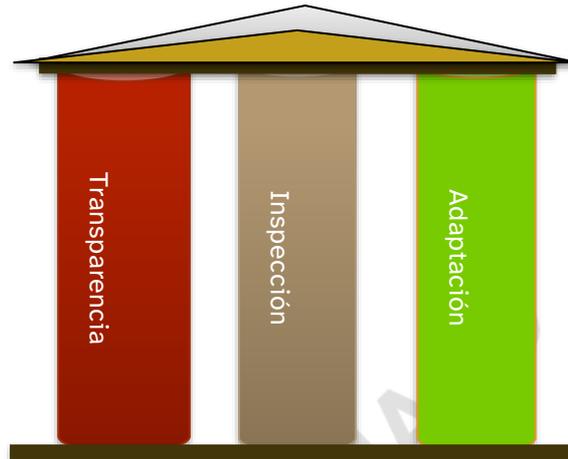
ARTEFACTOS

- Backlog del Producto
- Backlog del Sprint
- Definición de Hecho

Definición de Hecho

- Un conjunto de políticas acordadas para determinar si una tarea puede pasar de un estado del proceso a otro
- Establece un entendimiento común.
- Es una forma de comunicarse

Los tres pilares de Scrum:



1.2 Gestión con Scrum

Un Sprint es un espacio de tiempo predeterminado dentro del cual se genera valor.

El Product Owner (PO) es **responsable del éxito del proyecto** (la ejecución exitosa del proyecto y la creación exitosa es responsabilidad de todos los involucrados). El PO se comunica con el cliente para generar el backlog de producto y para priorizarlo en base a valor para el cliente. Las especificaciones son a grandes bloques con la definición suficiente para entender el alcance y la complejidad del proyecto, pero sin perder tiempo entrando en demasiado detalle que puede cambiar posteriormente o que sea demasiado temprano para saber los detalles.

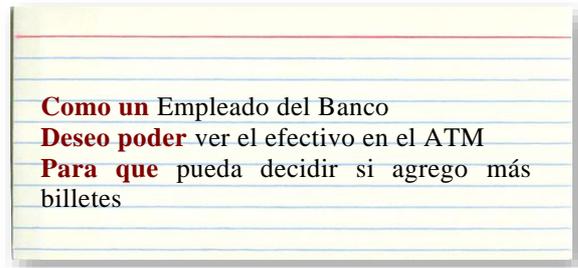
El PO debe mantener el backlog de producto siempre en buena forma tal que el trabajo por hacer pueda ser presentado en cualquier momento. A esa actividad de mantenimiento del backlog de producto se le llama **backlog grooming**. El Scrum Master y el equipo pueden colaborar con el PO para las actividades de grooming.

Backlog Grooming

- Lista ordenada de todo para el producto
- DEEP:
 - Detallado apropiadamente.
 - Estimado.
 - Emergente.
 - Priorizado.
- Administrado por el Product Owner.

La historia de usuario es un elemento esencial en ágil en general y en Scrum.

El alcance para generar el backlog tiene el formato de Historia de Usuario:



La historia de usuario tiene un formato predeterminado que elimina ambigüedad y a partir del cual las actividades de planificación, comunicación y seguimiento emergen. En la parte trasera de la tarjeta se especifican los criterios de aceptación

Al conformar el backlog de producto se lleva a cabo una **reunión de entrega** con todos los roles involucrados durante la cual se discute todo el backlog y se hace una estimación inicial con los objetivos de tener una idea general del alcance, generar alineamiento mediante las discusiones y hacer una estimación inicial de la cantidad de esfuerzo total.

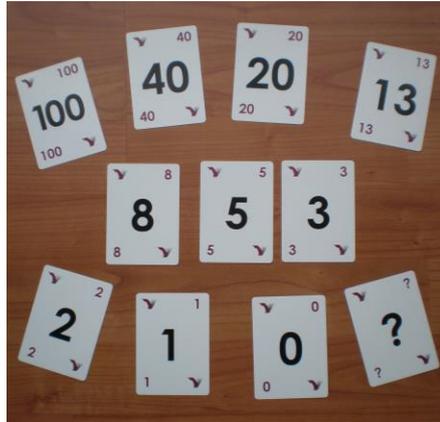
Una vez que el backlog está priorizado, el PO toma la parte de mayor prioridad y la detalla al nivel necesario para poder presentárselo al Scrum Master y al equipo de trabajo. El PO siempre debe llevarle al equipo más alcance del que ellos puedan llevar a cabo durante **una iteración de trabajo (el Sprint)**.

El Sprint es el ciclo de tiempo dentro del cual un alcance pequeño predeterminado debe ser completado. Por definición es de entre dos semanas a dos meses, aunque en la práctica tiende a no ser mucho más de dos semanas y raramente de más de cuatro semanas.

La reunión en la cual el PO muestra y explica las tareas al equipo para que se discutan y estimen para entonces determinar que tareas el equipo aceptará hacer durante el Sprint se llama reunión de **planificación del Sprint**. La planificación del sprint puede hacer uso de la herramienta **Póquer de planificación**. Nota: es posible utilizar el póquer de planificación en la planificación de la entrega.

El póquer de planificación es una herramienta para planificar y no una herramienta para estimar.

La estimación es un derivado de la planificación.



Es importante resaltar que, aunque la parte tangible de esta herramienta es la estimación lo más valioso es la planificación al llevar a cabo la discusión de las historias de usuario.

Las tareas que el equipo de trabajo acepta ejecutar durante el sprint se convierte en el **backlog del Sprint**. Durante el sprint el equipo trabaja de manera autoorganizada, con el Scrum Master (SM) dando apoyo y coordinando las actividades. El equipo y el SM se reúnen diariamente en la **reunión de Scrum** para mantener un ritmo sostenible y evitar que dependencias o bloqueos retrasen el trabajo. Mientras tanto el PO continúa mejorando el backlog de producto, comunicándose con el cliente y estando disponible para el equipo.

Al terminar el sprint, el equipo y el SM muestran al PO y al cliente el trabajo terminado en una **revisión del sprint**. Al término de esta actividad el PO acepta o rechaza el trabajo presentado. Se procede entonces a hacer la **retrospectiva del sprint** donde se determina, a nivel del proceso, qué se ha hecho bien para seguir haciéndolo, qué se ha hecho mal, para dejar de hacerlo, y qué mejorar.

Tablero de Scrum

Por hacer	Haciendo	Hecho

2 Kanban

2.1 Las seis prácticas de Kanban.

Kanban es un método enfocado en la **mejora continua** de la organización mediante la mejora de flujo con énfasis en su economía. Es importante notar que el sistema kanban de manufactura y el método Kanban de trabajo de conocimiento son distintos y tienen orígenes independientes. El sistema kanban es parte de lean de manufactura. El método Kanban se inició en el año 2004 y consiste en **seis prácticas** [Anderson].

2.1.1 Alta visualización.

Supongamos que tenemos un proceso dentro del cual se trabajan distintos tipos de tareas, por ejemplo, las características del producto solicitado por el cliente, trabajo interno del equipo y tareas de infraestructura.

El primer tablero muestra la cantidad total de tareas, cuántas están en progreso y cuantas terminadas, pero nada más.

Por Hacer	Haciendo	Hecho
		
		
		
		

El siguiente tablero muestra a detalle lo que se está haciendo y eso nos da un mayor entendimiento del estado del proceso.

Por Hacer	Diseñar	Construir	Verificar	Entregar
				
				
				
				
				

Kanban es un método para mejorar lo que tenemos.

La alta visualización incrementa el entendimiento.

El tercer tablero incluye subcolumnas y los distintos tipos de tareas están diferenciados por colores. Esto **elimina la ambigüedad** de saber si alguna tarea está siendo trabajada o si está esperando para ser tomada por la siguiente etapa del proceso. Los colores nos permiten saber el monto de trabajo para cada tipo de tarea y su estado dentro del proceso. Esta alta visualización **incrementa la efectividad del equipo y mejora el flujo** porque permite tomar decisiones más pertinentes que cuando no tenemos este nivel de entendimiento.

Por Hacer	Diseñar		Construir		Verificar		Entregar
	haciendo	hecho	haciendo	hecho	haciendo	hecho	
 	 		 				   

2.1.2 Limitar la cantidad de trabajo en progreso.

Lean nos muestra que si a un recurso (persona, personas, maquinaria, o combinaciones de ellas) se les entrega mucho trabajo por hacer el resultado lleva mucho tiempo, tiene baja calidad e incrementa el nivel de estrés. Si en lugar de eso, les entregamos una cantidad pequeña de trabajo a la vez, el resultado será que terminarán la **misma cantidad de trabajo, en menor tiempo, con mejor calidad y menor nivel de estrés.**

Limitar el WIP (work in progress) mejora la economía del sistema.

La tabla ilustra esto. Nótese como para seis equipos la cantidad de tiempo que les lleva terminar 12 tareas es mucho mayor cuando tienen las 12 tareas en un solo lote comparado con tener dos lotes (trabajando solo un lote a la vez) o cuatro lotes, etc. Esto se debe a que existe desperdicio en cuanto al tiempo de transición de las tareas y el tiempo de espera de las tareas. Adicionalmente, cuando el trabajo es demasiado las personas reaccionan con estrés.

	Monto de trabajo por lote				
Equipo	(1) 12	(2) 6	(4) 3	(6) 2	(12) 1
Equipo 1 tiempo	4m 15s	2m 45s	1m 48s	1m 29s	1m 16s
Equipo 2 tiempo	4m 17s	2m 2s	1m 22s	1m 15s	1m 15s
Equipo 3 tiempo	5m 7s	2m 59s	2m 14s	1m 54s	1m 41s
Equipo 4 tiempo	4m 4s	2m 8s	1m 31s	1m 34s	1mm43s
Equipo 5 tiempo	4m 54s	2m 27s	1m 36s	1m 34s	1m 23s
Equipo 6 tiempo	4m 8s	3m 9s	1m 55s	1m 52s	1m 18s

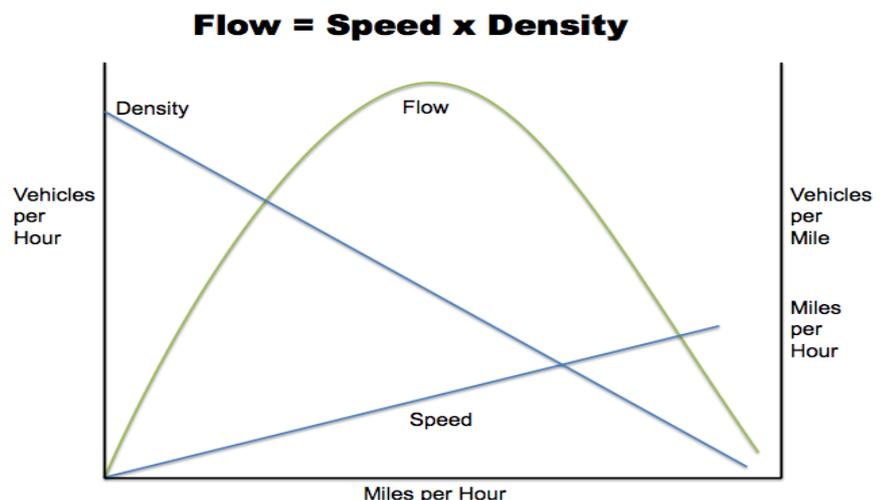
El límite de trabajo en progreso se muestra en Kanban explícitamente en el encabezado de las secciones del tablero donde sean aplicables.

Por Hacer	Diseñar 2		Construir 4		Verificar 2		Entregar
	haciendo	hecho	haciendo	hecho	haciendo	hecho	

Se recomienda también que el tablero tenga una zona dedicada a tareas express.

2.1.3 Mejorar colaboradoramente y evolucionar experimentalmente

Kanban nos indica que utilicemos modelos y el método científico. Kanban mismo no cuenta con modelos, pero sugiere algunos además de dejar carta abierta para aplicar otros. El modelo de flujo es un ejemplo útil y fue propuesto por [Reinertsen].

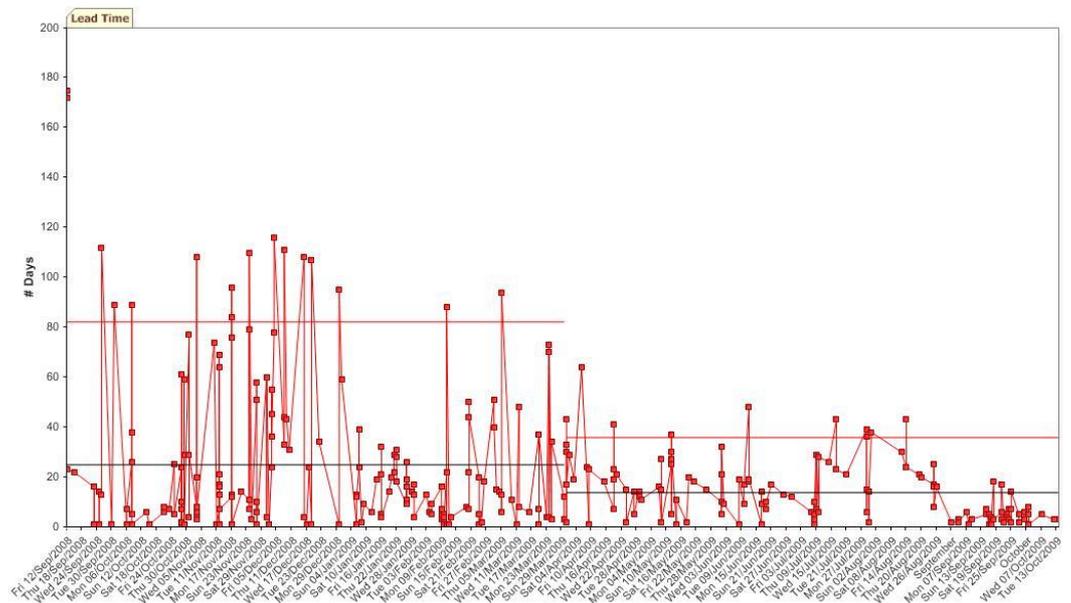


El uso de modelos es esencial para la gestión cuantitativa.

Este modelo nos dice que la densidad (el monto de trabajo en progreso) resulta en un cierto monto de flujo y que para dos densidades distintas podemos obtener el mismo nivel de flujo, la diferencia radica en que para un monto de trabajo bajo el sistema es estable (mejor economía) comparado con el monto alto que es inestable y tiene un coste económico mayor.

2.1.4 Gestionar el flujo.

La gráfica nos muestra el diagrama de control para el caso real en una empresa líder en su industria.



Gestionar el flujo incrementa la satisfacción del cliente.

El eje horizontal son fechas de calendario y el eje vertical es duración en días. Cada punto rojo nos indica una entrega, en qué fecha fue entregada y cuánto tiempo le tomó terminar. El patrón evidente es que conforme avanzó el proyecto el monto de tiempo requerido para terminar las tareas fue disminuyendo. Eso no es una coincidencia sino **el resultado de hacer mejora continua aplicando las prácticas**. Conforme las tareas se llevan a cabo son analizadas para determinar cómo mejorar el flujo eliminando variaciones negativas.

2.1.5 Hacer los procesos explícitos.

Una forma de hacer más explícitos los procesos, en adición a tarjetas de colores (clases de servicio), es generar **políticas acordadas de comportamiento** para cada color, por ejemplo:

Ser explícitos evita que se genere desperdicio y mejora la colaboración

-  **Tarea Normal**
 - No se estima
 - Se trabajan en el orden de entrada
 - Se entregan al final del día

-  **Trabajo Interno**
 - ◆ No se estima
 - ◆ Se trabajan en el orden de entrada
 - ◆ No hay otra en Diseño

-  **Fecha Fija**
 - ◆ Se estima
 - ◆ Entra dos días previos al tiempo de estimación
 - ◆ Mayor prioridad que Normal e Interno

Las políticas explícitas no solo **mejoran la colaboración**, también **eliminan ambigüedad** y hacen de la toma de decisiones parte del método en lugar de una actividad basada en juicio experto o en impulso.

2.1.6 Implementar ciclos de retroalimentación

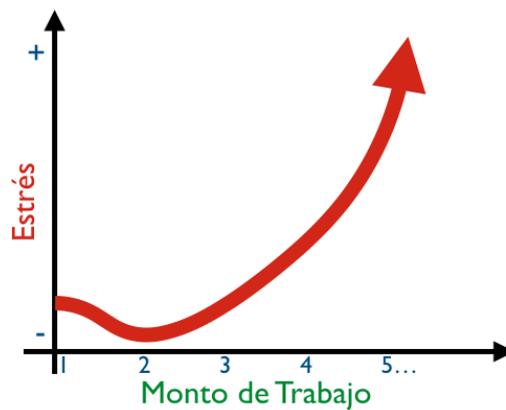
Kanban recomienda llevar a cabo reuniones diarias basadas en el flujo de valor. Es decir, en lugar de tener reuniones centradas en la persona, como lo hace Scrum, el equipo completo se reúne ante el tablero de Kanban para discutir el progreso diario. Las ventajas de hacer esto radican en que la cantidad de tiempo total de la actividad es menor que el equivalente con Scrum, se pueden tener grupos grandes de personas en la misma reunión **y todos los involucrados saben el estado del proyecto.**

Alineamiento diario dentro del alcance del tablero Kanban.

2.2 Gestión con Kanban

La gestión con Kanban se facilita si las seis prácticas son aplicadas. Cuando el progreso diario se visualiza, se siguen las políticas y se hace cuantificación, la gran mayoría de la toma de decisiones se basará en entendimiento y las acciones tendrán menos riesgo.

Se debe considerar siempre el aspecto humano. La gráfica muestra la curva de estrés con respecto a la cantidad de trabajo en progreso.



Es importante que **el tablero de Kanban refleje la realidad del proceso**. Es por ello que los tableros pueden tomar cualquier forma, por ejemplo:

El aspecto humano siempre se debe considerar.

	Requests	Ready	In Progress	Customer Verification	Done :-)
Project A	3	2	3	2	4
Project B	4	2	2	1	3
Sales	2	2	5	3	2
Financial	1			1	
Customer X	3	2	2	2	5
Customer Y	5	2		3	2
Expedite +!			1		

Con Kanban logramos previsibilidad.

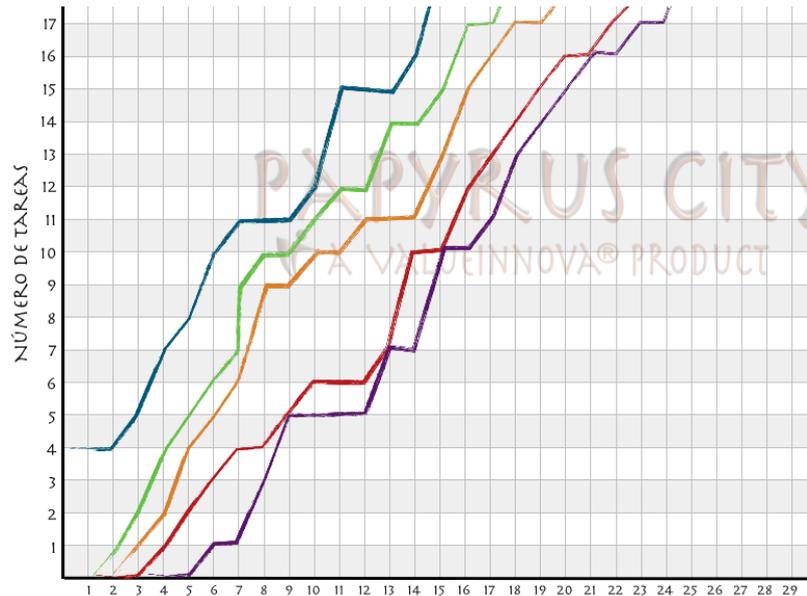
Con Kanban es posible **identificar cuellos de botella aún antes de que sucedan**. El Tablero muestra un estado de acercamiento a un cuello de botella que se puede observar visualmente por la zona con alta densidad de tarjetas en la parte derecha y de baja densidad en la parte izquierda.

Diseñar 2		Resolver 3		Verificar Int 4		Verificar Ext 3		Entregar
doing	done	doing	done	doing	done	doing	done	
1		1	1	1	1	1	1	1
			1		1	1	1	1
					1	1		1
					1			1

La siguiente figura muestra el estado de cuello de botella.

Diseñar 2		Resolver 3		Verificar Int 4		Verificar Ext 3		Entregar
doing	done	doing	done	doing	done	doing	done	
	1		1		1	1		1
			1		1	1	1	
			1		1	1	1	
			1		1			

El diagrama de flujo acumulado refleja el avance de las tareas en el tablero donde cada línea de izquierda a derecha corresponde a cada columna del tablero también de izquierda a derecha.



Nótese como cuando un cuello de botella puede llegar a existir, o llega a existir, el espacio entre líneas se incrementa como se ve entre las líneas naranja y roja.

Como parte de las políticas explícitas podemos tener el establecimiento de cadencias donde, por ejemplo, cada tipo distinto de tarea (clase de servicio) tiene un periodo distinto de entrega basada en la necesidad del cliente y en la capacidad de satisfacer esa necesidad.

2.3 Kanban personal.

Kanban personal se enfoca en formas en las que Kanban puede ser utilizado efectivamente a nivel individual. Este trabajo [Benson] obtuvo el premio Shingo en el año 2015. Jim Benson indica que “No podemos tomar decisiones informadas o crear un producto de calidad sin primero entender por qué estamos haciendo lo que hacemos. La carencia de contexto genera desperdicio, resultando en días largos de trabajo, mala planificación y la incapacidad de mantener compromisos fuera de la oficina”.

Kanban personal tiene dos reglas:

1. Visualizar tu trabajo.
2. Limitar tu monto de trabajo en progreso.

Esto permite mejor gestión del trabajo por hacer además de permitir hacer mejora continua a nivel individual. Intentar gestionar el trabajo individual sin elementos externos tangibles:

- Dificulta la comprensión del porqué de nuestra situación actual por lo que es difícil mejorar.

Con Kanban logramos

- El aprendizaje se dificulta.
- Es muy difícil identificar patrones de comportamiento.
- Tendemos naturalmente a tener demasiadas cosas por hacer sobre nuestros hombros.

Para generar tu tablero de Kanban personal:

- Haz tu proceso explícito.
- Escribe las tareas que ya tienes.
- Establece tu límite de trabajo en progreso. Un número recomendable es 2 o 3. Uno te pondrá en problemas y te estresará.
- Trabaja en base a una modalidad de arrastre.
- Reflexiona sobre el comportamiento resultante para analizar como mejorar tu flujo.

3 Scrumban

Scrumban es, como su nombre lo sugiere, un híbrido entre Scrum y Kanban. Scrum tiene la bondad de ayudar a generar una buena disciplina de trabajo y una expectativa fija de entrega. Kanban tiene la bondad de fomentar una cultura de mejora continua y gestión cuantitativa. Scrumban es apta para organizaciones donde ambas bondades son necesarias.

El Scrumban típico consiste en un tablero de Scrum al cual se le agregan límites de trabajo en progreso y visualización de tipos de tareas utilizando etiquetas de distintos colores. Una extensión es detallar el tablero para poder establecer límites de trabajo en progreso mejor e inclusive generar políticas ligeras. Una buena implementación de Scrum cuenta ya con políticas tales como la Definición de hecho.

El Tablero mostrado es un ejemplo de un Scrumban. Sprints y backlogs se siguen utilizando.

Agregarle bondades de Kanban a Scrum.

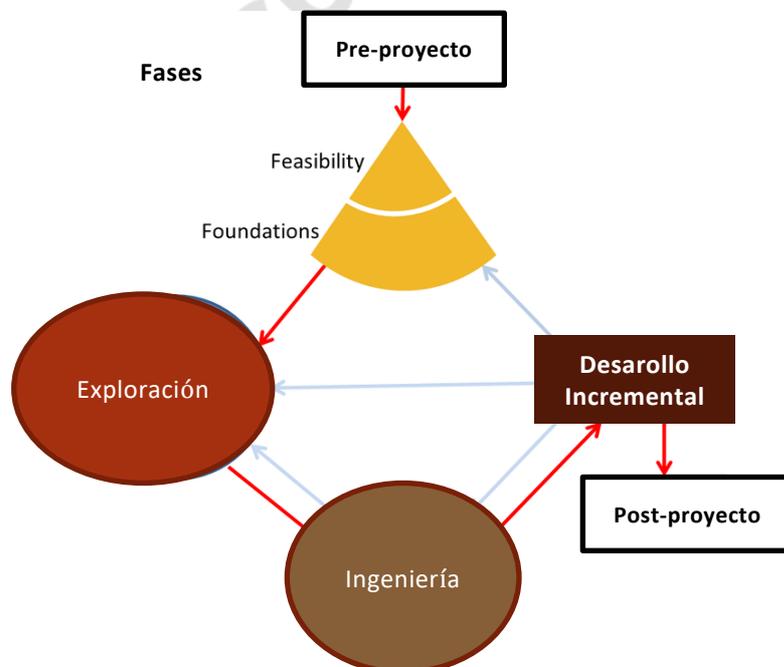
Por Hacer	Diseñar 2	Construir 3	Verificar 2	Entregar
				
				
				

4 Otras Metodologías

4.1 DSDM

DSDM son las iniciales del Dynamic Systems Development Method. Fue una evolución ágil a partir del método RAD (rapid application development). Es más elaborado que Scrum y Kanban y su uso en Europa es significativamente mayor que en el resto del mundo. Es **el método ágil más cercano a la gestión tradicional** por lo que algunas personas lo utilizan para proyectos muy grandes.

Considera costo, tiempo y calidad como constantes y opera junto con otras estructuras tales como ITIL, PRINCE2® y PMBOK(r).

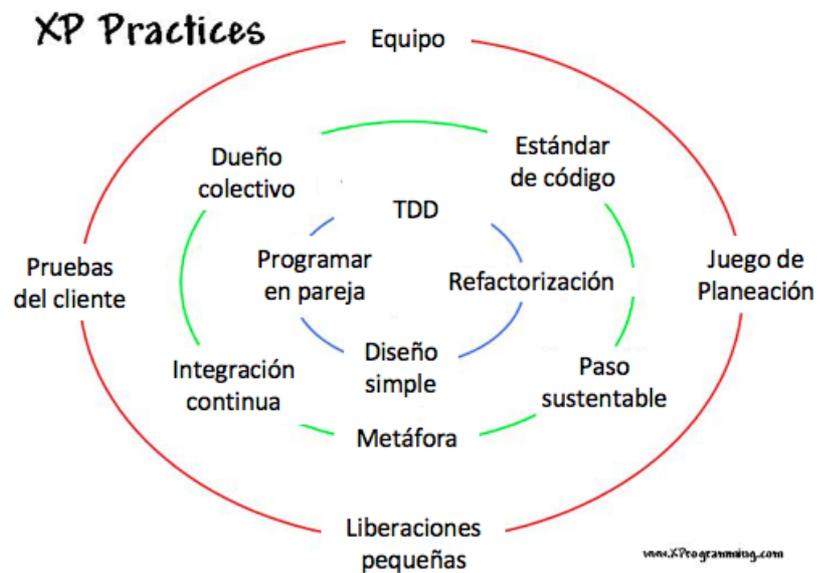


4.2 Programación extrema

Mejor conocido como XP, la programación extrema fue desarrollada por Ron Jeffries y Kent Beck. Está enfocado en el desarrollo de software y, contrario a lo que se podría esperar, su base no es técnica sino humana.

El equipo está conformado por todos los miembros del proyecto, quienes orbitan alrededor del cliente. Sus prácticas están organizadas en tres capas.

XP se enfoca en desarrollo de software.



La capa externa tiene que ver con las actividades de contacto con el mundo exterior. La capa interna tiene que ver con las actividades técnicas y la capa intermedia tiene que ver con prácticas que interconectan las otras dos.

Valores

- Comunicación
- Retroalimentación
- Respeto
- Coraje
- Simplicidad

Principios

- Humanidad
- Diversidad
- Beneficio mutuo
- Economía
- Flujo
- Calidad
- Pasos pequeños
- Retroalimentación
- Reflexión
- Responsabilidad
- Oportunidad
- Falla
- Redundancia
- Auto similitud
- Mejoramiento

5 Escalado de Ágil

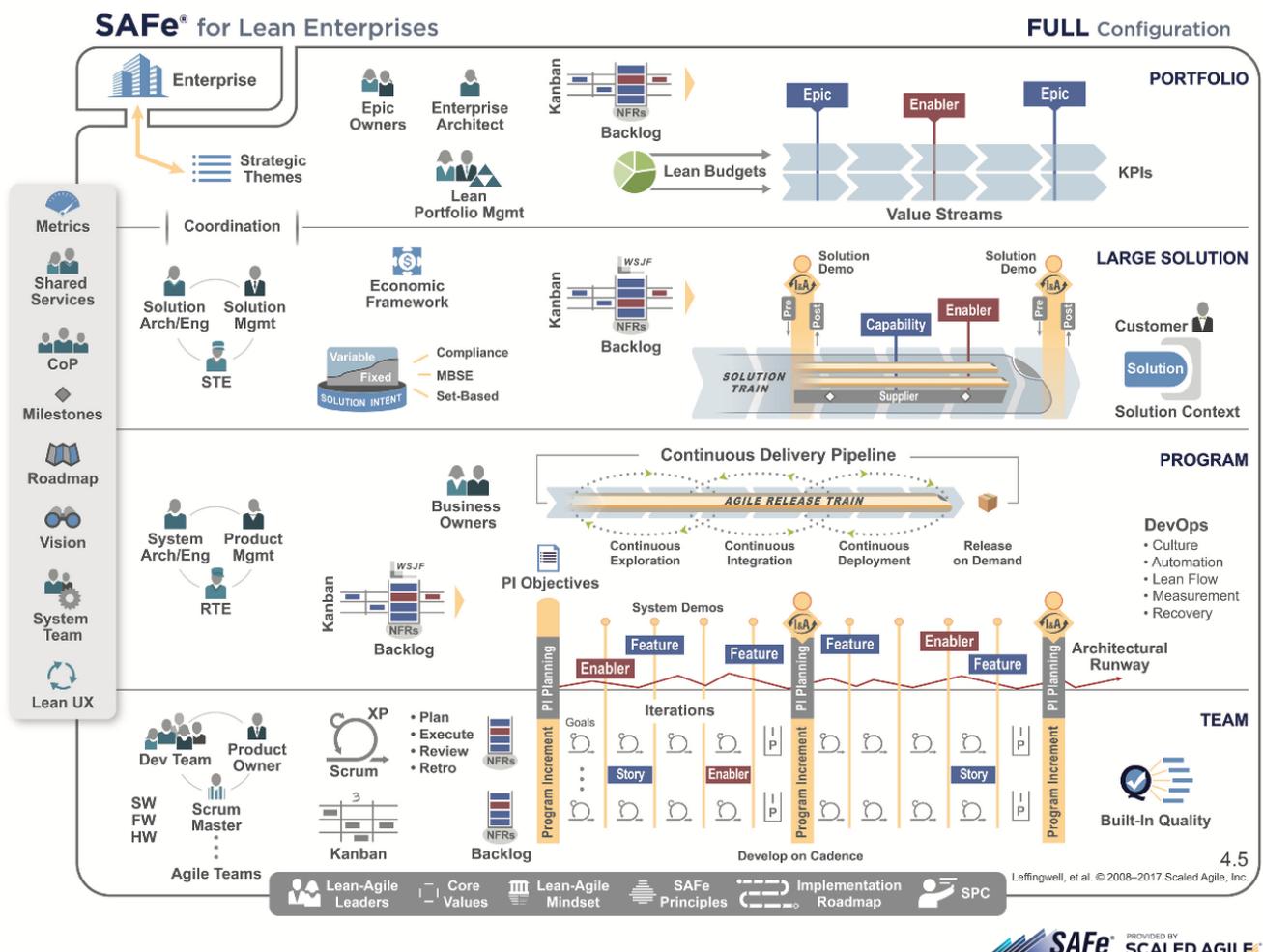
5.1 SAFe

SAFe es para empresas grandes y proyectos grandes.

SAFe (Scaled Agile Framework) es una estructura desarrollada por Dean Leffingwell para aplicar ágil y lean—mucho más Scrum en su versión 1 y gradualmente más Kanban en versiones recientes, a nivel empresarial y para proyectos grandes.

La estructura y nomenclatura es cercana a los enfoques tradicionales lo cual ha facilitado su aceptación en ambientes empresariales. Ha madurado para tener 4 versiones: Completo, portafolio, soluciones grandes y esencial.

La estructura es completa es:



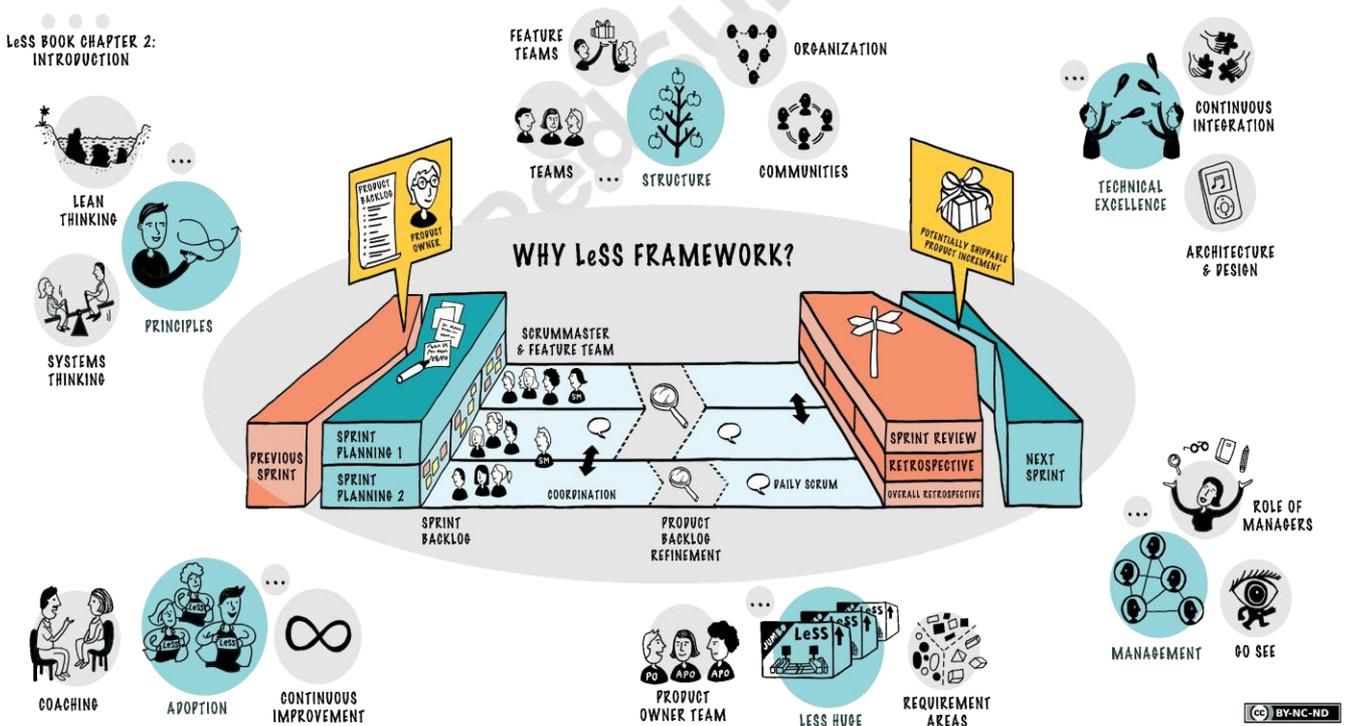
Como se puede observar, consiste en cuatro capas, desde nivel estratégico (portafolio) a táctico (solución grande) y operativo (programa y equipo). Nótese también la integración de estructuras y metodologías lean-agile (Kanban, Scrum, XP).

Al nivel más alto se trabajan épicas mediante Kanban para seguir el flujo de valor identificado. A partir del backlog de portafolio se generan soluciones grandes con un Kanban para cada solución y son organizados en trenes de solución. En la siguiente capa, programa, se hace la gestión del producto también en base a Kanban y donde los trenes operan bajo exploración-integración-entrega continua. A nivel del equipo se opera en base a Scrum y/o Kanban en interacción continua y cercana con la capa de programas.

SAFe es la estructura ágil de escala más popular.

5.2 LESS

La segunda estructura más utilizada para el escalado ágil es LESS (Large Scale Scrum) [LESS]. Fue desarrollada por Bas Vodde y Craig Larman.



LESS es más ligero que SAFe.

LESS está centrado en Scrum.

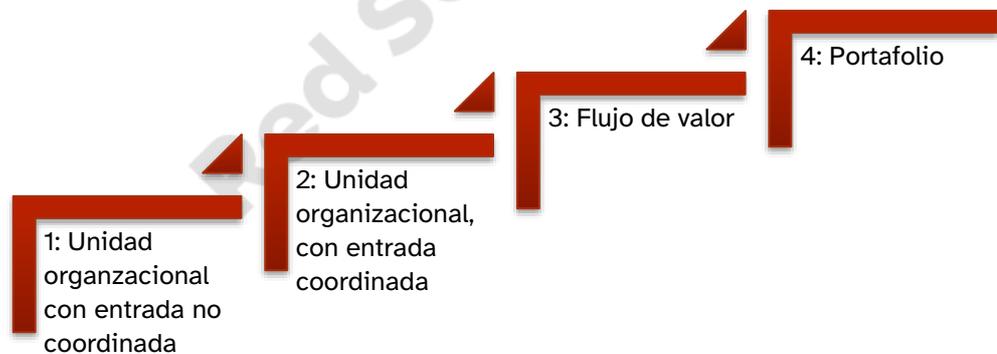
LESS está centrado en Scrum. Consiste en una serie de principios donde persigue **simplificar lo que genera sobrecarga al escalar naturalmente**. Promueve lograr más con menos tal que la simplicidad persista, más esfuerzo se enfoque en genera valor y no en coordinar, reducir segregación, mantener el proceso simple, incrementar flexibilidad y reducir inventario.

Su naturaleza puede hacerlo menos atractivo para ejecutivos de megaempresas inclinados a procesos detallados porque no tiene la estructura compleja que ofrece SAFe. Para ejecutivos con enfoque en mantener una competitividad alta y avanzar rápidamente sin incrementar costos y complejidad, LESS puede ser más atractivo.

5.3 Niveles de vuelo.

Esta sección debería llamarse, si fuésemos canónicos, los niveles de vuelo de Kanban. Sin embargo, su carácter agnóstico permite verlo independientemente del contexto de Kanban.

Klaus Leopold [LEOPOLD] sugirió esta estrategia para escalar el impacto de Kanban naturalmente dentro de la organización.



El primer nivel se enfoca en un núcleo (equipo) de manera hermética donde la mejora es interna. Esto implica que ese núcleo sigue interactuando con externos como de costumbre, lo cual obviamente limita el nivel de mejora que se puede

lograr, pero tiene el beneficio de que se adquiere disciplina y madurez interna lo cual les prepara para el siguiente nivel.

En el nivel dos el equipo influye en sus interfaces no solo a nivel de lo que reciben y entregan sino en la forma en la que esas interfaces operan dentro de ese contexto. En el nivel tres se mejora toda la cadena de valor relacionada con ese equipo o proceso. El cuarto nivel es, una vez que la cadena de valor ya opera bajo lean y Kanban, extender la experiencia al resto de las cadenas de valor de la organización.

6 Conclusiones

- Scrum es una estructura y Kanban es un método.
- Scrum reemplaza el estilo de gestión.
- Kanban no reemplaza a nada. Kanban mejora lo que ya se tiene.
- Scrum introduce roles, eventos y artefactos.
- Kanban preserva los roles existentes.
- Las Historias de Usuario se pueden utilizar en cualquier estructura y metodología lean o ágil.
- El póquer de planificación es para planificar no para estimar. La estimación es un derivado.
- El método Kanban consiste en seis prácticas.
- Limitar adecuadamente el monto de trabajo en progreso (WIP) reduce el estrés.
- Scrum utiliza gestión cualitativa, Kanban utiliza gestión cuantitativa.
- Kanban nos da previsibilidad.
- Kanban personal ayuda a mejorar nuestra vida personal y laboral.
- Scrumban combina las bondades de Scrum con algunas de las bondades de Kanban.
- XP tiene que ver más con el aspecto humano que con técnica.
- SAFe y LESS son enfoques muy distintos para escalar ágil.

- Los niveles de vuelo son agnósticos.

7 Bibliografía

- [Anderson] David Anderson. Kanban: Successful Evolutionary Change for Your Technology Business. Blue Hole. 2010.
- [Benson] Jim Benson. Personal Kanban. Modus Cooperandi Press. 2011.
- [Beck] Kent Beck. Extreme Programming Explained: Embrace Change, 2nd ed. Addison-Wesley. 2004.
- [Burrows] Mike Burrows. Kanban From the Inside. Blue Hole Press. 2014.
- [Leffingwell] SAFe 4.0 Reference Guide: Scaled Agile Framework for Lean Software and Systems Engineering. Addison-Wesley. 2016.
- [LESS] <https://less.works/>
- [Reinertsen] Donald Reinertsen. Principles of Product Development Fow. Celeritas Publishing. 2009.
- [SAFe] <http://www.scaledagileframework.com/>
- [Sutherland] Jeff Sutherland. Scrum: The Art of Doing Twice the Work in Half the Time. Crown Business. 2014.
- [Takeuchi & Nonaka] Donald Reinertsen. Principles of Product Development Fow. Celeritas Publishing. 2009.