

INTRODUCCIÓN A LAS METODOLOGÍAS ÁGILES

**MÓDULO PROFESIONAL:
FULL STACK**



Índice

RESUMEN INTRODUCTORIO.....	3
INTRODUCCIÓN	3
CASO INTRODUCTORIO	3
1. INTRODUCCIÓN A LA PLANIFICACIÓN DE PROYECTOS	5
1.1 Modelo general de proyecto.....	5
1.2 Tipos de ciclos de desarrollo	6
2. METODOLOGÍAS ÁGILES CONTRA NO ÁGILES.....	8
2.1 El manifiesto ágil	10
2.2 Cuando usar metodologías ágiles. Desarrollo de software.....	10
3. CONCEPTOS BÁSICOS SOBRE SCRUM	12
3.1 Scrum técnico.....	14
3.2 Artefactos	15
3.3 Eventos	15
3.4 Roles	15
3.5 Historias de usuario.....	16
4. CONCEPTOS BÁSICOS SOBRE KANBAN.....	17
4.1 KANBAN y SCRUM	18
4.2 Trello.....	19
4.2.1 Comenzando a usar Trello.....	19
4.2.2 Primer tablero y tareas	20
4.2.3 Gestión colaborativa	20
RESUMEN FINAL.....	22

RESUMEN INTRODUCTORIO

En esta unidad introduciremos los conceptos básicos sobre los proyectos de software y en concreto sobre la planificación y metodologías que podemos usar para la poder llevar a cabo una aplicación o un servicio dentro de un equipo de desarrollo.

En concreto, nos centraremos sobre todo en las metodologías ágiles, como SCRUM, ya que son conocimientos clave para poder comenzar a trabajar dentro de un equipo de desarrollo hoy en día, definiendo los conceptos clave y aportando una visión global y visual de esta forma de trabajar.

Por último, introduciremos otra metodología que combina perfectamente con SCRUM como es KANBAN. Una de las herramientas más usadas y conocidas hoy en día para llevar a cabo KANBAN es Trello, por ese motivo daremos las pautas para sus conceptos clave.

INTRODUCCIÓN

Hoy en día, vayamos a gestionar un proyecto o vayamos a trabajar dentro de un equipo de trabajo, es clave conocer las diferentes metodologías de trabajo que las empresas aplican para el desarrollo del software. Esta gestión y trabajo ha evolucionado enormemente en los últimos tiempos, en la misma medida que el trabajo en sí.

Nos encontramos con equipos de trabajo cada vez más heterogéneos y deslocalizados que deben conocer el proyecto en su globalidad y tener la responsabilidad suficiente para poder llevar a cabo un proyecto en tiempo y en forma. Por ese motivo las personas, cada vez más, pasan a un primer plano, por ese motivo las metodologías ágiles han tomado tanta fuerza.

Scrum se ha convertido en prácticamente un standard dentro de la empresa. Su adaptación a los proyectos, su facilidad para dar responsabilidad al equipo de trabajo, la importancia que se le da a que el cliente final, sea externo o interno a la empresa, esté relacionado con el proyecto, le ha hecho convertirse en una metodología necesaria conocer y usar.

Muchas son las herramientas tecnológicas de gestión de proyectos que han evolucionado hacia esa posibilidad de incluir la terminología y filosofía Scrum, una de ellas Trello, basada en otra técnica muy visual como es KANBAN.

CASO INTRODUCTORIO

Te han contratado como desarrollador junior dentro de una empresa de desarrollo de software. En la empresa, y en concreto en el departamento donde trabajarás, se aplica la

metodología SCRUM usando, además, herramientas como Trello para la autogestión de las tareas y las fases del proyecto.

Al incorporarte dentro de este modelo de trabajo, necesitas conocer los conceptos básicos para comenzar a desarrollar y participar en las tareas que se deben finalizar. En concreto te han contratado para trabajar en un nuevo proyecto para el desarrollo de una aplicación web la cual recogerá la asistencia a eventos.

Al final de esta unidad conocerás las características de la metodología SCRUM y su combinación con KANBAN en la gestión de proyectos mediante el uso de herramientas como Trello.

1. INTRODUCCIÓN A LA PLANIFICACIÓN DE PROYECTOS

Antes de comenzar tu trabajo y empezar a asumir tareas del proyecto, quieres conocer los fundamentos sobre la gestión de proyectos y los ciclos que intervienen en los mismos, así como el sistema de gestión de trabajo que lleva a cabo la empresa. Por ello, decides investigar un poco sobre la terminología básica de la gestión de proyectos y los tipos de proyectos y de gestión existentes.

La planificación de proyectos es uno de los aspectos más interesantes y a la vez complejos en el desarrollo de software. De hecho, hay toda una disciplina denominada Ingeniería del Software, dedicada a los aspectos de la producción del software, desde las etapas más tempranas de la recogida de las especificaciones del desarrollo dentro de la planificación de proyectos, hasta el mantenimiento de las aplicaciones pasando por la producción y gestión de proyectos.

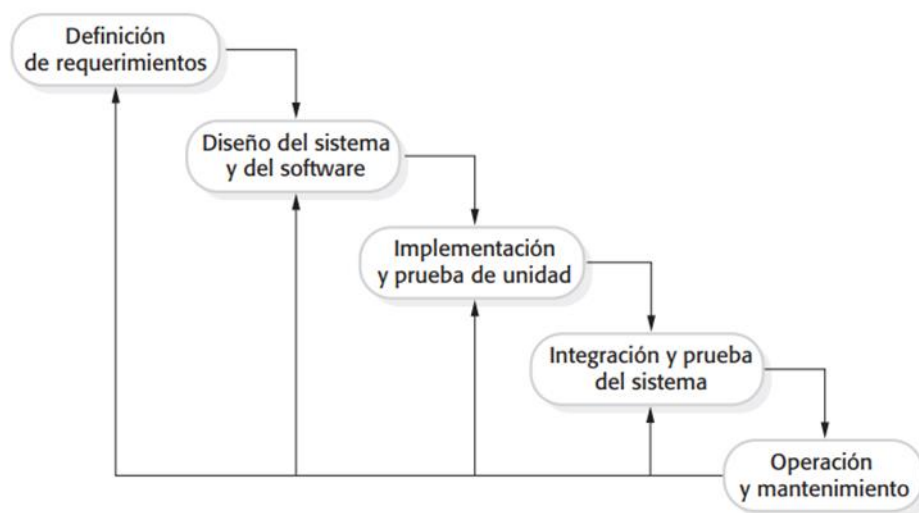
Un proceso de desarrollo del software, estemos planteando una gestión ágil o más secuencial podemos decir que pasa por las siguientes fases:

- Especificación del software, esto es la recogida de las funcionalidades de la aplicación a realizar.
- El diseño e implementación del software, orientado a cumplir con las especificaciones recogidas.
- Validación para asegurarse que se cumple con las especificaciones.
- Pruebas y mantenimiento, orientado a la evolución del software una vez instalado.

No hemos querido marcar una numeración y un orden, ya que tal y como veremos justamente en esta unidad didáctica, hoy en día estos apartados pueden alterarse o producirse en varios puntos del desarrollo.

1.1 Modelo general de proyecto

Un proyecto lo podemos definir como el conjunto de actividades, de acciones y tareas que se realizan cuando va a crearse algún desarrollo en concreto. Cada una de las actividades, acciones y tareas se encuentra dentro de un modelo que están relacionados tanto con el resto de las partes como consigo mismo. La imagen siguiente representa el proceso del desarrollo software de forma esquemática.



Proceso lineal

Fuente: Sommerville, I (2011). INGENIERÍA DEL SOFTWARE (9.a ed.)

Como vemos, una estructura general define cinco actividades estructurales:

- Comunicación.
- Planeación.
- Modelado.
- Construcción.
- Despliegue.

Aunque esta clasificación dependerá muy mucho de la tipología y objetivos del proyecto. Además, dentro de cada proceso o tarea, podemos modelarlo por: seguimiento y control del proyecto, administración de riesgos, aseguramiento de la calidad, administración de la configuración, revisiones técnicas, y otras muchas.

1.2 Tipos de ciclos de desarrollo

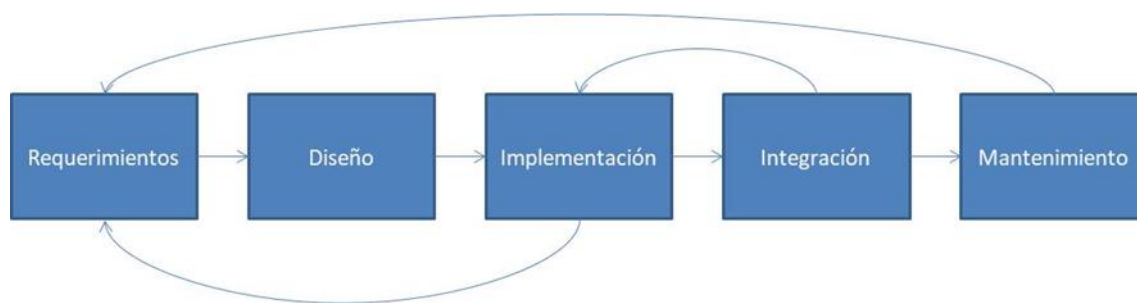
Lo que sí que es interesante es tener una idea gráfica de diferentes modelos de ciclos de desarrollo, y que posteriormente veremos cómo se ligan a diferentes modelos de planificación de proyectos.



Proceso lineal.

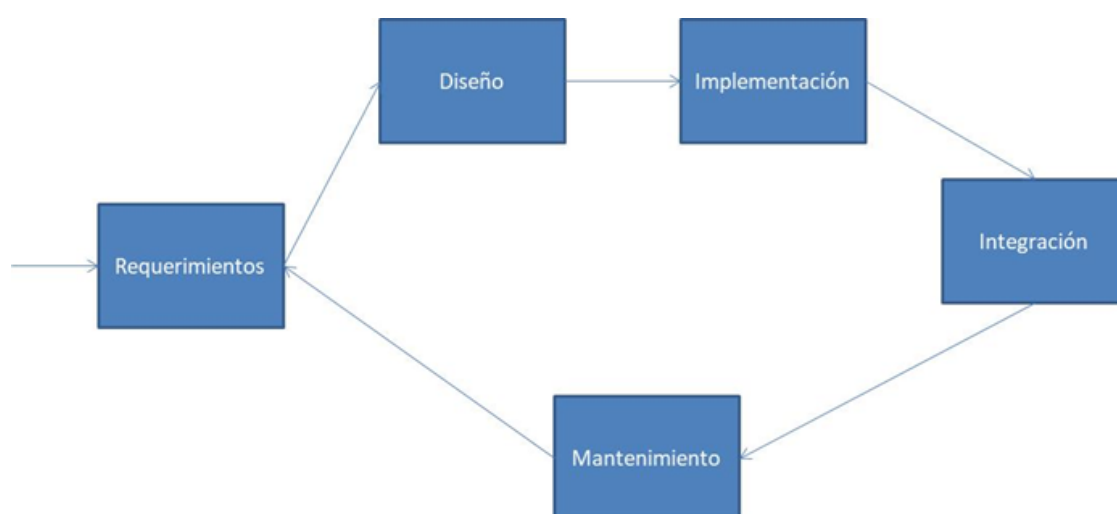
Fuente: Imagen propia

Como vemos en la anterior imagen, cuando hablamos de proyectos, quizá el modelo más básico es el de un flujo lineal, donde las fases se producen una detrás de otra. Veremos enseguida que este tipo de flujos y gestión no siempre es la más adecuada.



Proceso iterativo.
Fuente: Imagen propia

Una evolución del anterior modelo es el iterativo, donde parte de las fases se iteran hasta conseguir el resultado planificado.



Proceso evolutivo.
Fuente: Imagen propia

Por último, un modelo evolutivo, más cerca de las metodologías ágiles, planifica sus tareas de una forma circular, a lo largo de las 5 fases propuestas o de parte de ellas.

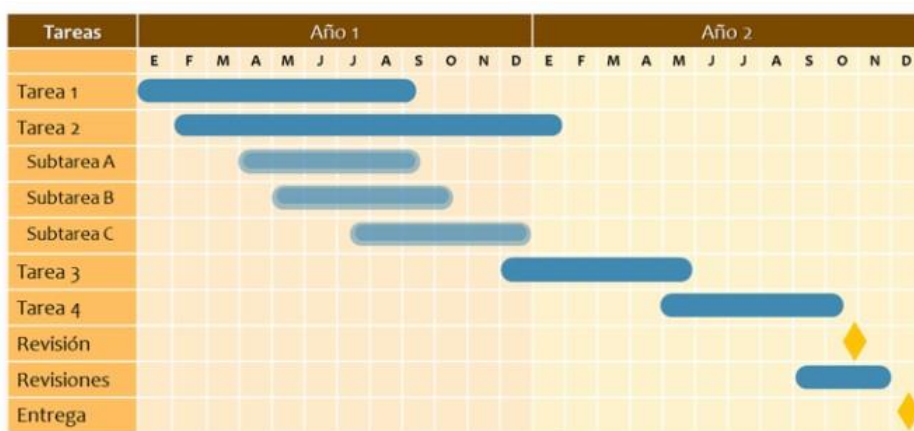
2. METODOLOGÍAS ÁGILES CONTRA NO ÁGILES

Dentro del manual y actividades de “onboarding” de la empresa aparecen muchos conceptos relacionados con la gestión de proyectos.

En concreto, al comenzar tu primer proyecto dentro de un equipo de trabajo donde se usan metodologías ágiles te preguntas qué diferencia existe entre las metodologías ágiles y no ágiles, así como si se pueden usar las metodologías ágiles en todos los tipos de proyectos.

La actualidad en el desarrollo de proyectos se parece muy poco al modelo de gestión de proyectos de hace diez o veinte años dentro del desarrollo de software. Ahora mismo necesitamos nuevos modelos y estrategias para la entrega del producto por partes, una respuesta **ágil** y **flexible**. Todo ello debido a que el mercado actual se mueve a una velocidad mucho más rápida.

Cuando pensamos en gestión de proyectos seguro que nos viene a la cabeza el típico diagrama de Gantt, donde las actividades se piensan y se preparan de una forma secuencial, una tras otra, con un coste, con un responsable. Pertenecen esta gestión a la llamada metodología predictiva, un modelo de gestión de proyectos que plantea esta como el uso más eficiente de los recursos para cumplir unos plazos, costes y calidad. Es decir, una gestión basada en una fuerte definición de las especificaciones funcionales y no funcionales.



Ejemplo Diagrama de Gantt.

Fuente: <https://templates.office.com/es-es/diagrama-de-gantt-de-dos-a%C3%B1os-tm56599548>

Sin embargo, nos podríamos plantear las siguientes preguntas ¿La gestión predictiva es la única que podemos aplicar a los proyectos? ¿El éxito en la gestión de un proyecto es siempre el cumplimiento de las fechas y los costos? ¿Existen proyectos que no se gestionen planificando absolutamente todo el proyecto a partir de unas especificaciones?

La respuesta ni es absoluta ni siempre es cierta para todos los proyectos. Hay proyectos donde se necesita tener de una forma rápida unos mínimos para poder realizar una prueba de concepto o de usabilidad, sin haber finalizado el proyecto completo. Hay proyectos que

cambian las especificaciones durante el mismo proyecto. Y hay proyectos donde la metodología predictiva sigue siendo igual de válida.

Podemos realizar una primera tabla donde comparemos ambas metodologías, para ver de un vistazo donde podríamos aplicarlo:

	METODOLOGÍAS PREDITIVAS	METODOLOGÍAS ÁGILES
PROS	<ul style="list-style-type: none"> Mejor para proyectos para el desarrollo de un producto físico. Cableado estructural, por ejemplo. Mejor para proyectos grandes o gubernamentales. Mejor para proyectos repetitivos 	<ul style="list-style-type: none"> Se adaptan muy bien a los proyectos no físicos, software o de servicios. Se adapta muy bien a equipos multidisciplinares. Fomenta la responsabilidad entre los miembros del equipo
CONTRAS	<ul style="list-style-type: none"> El tiempo invertido en la creación de las especificaciones es muy grande. No se adaptan nada bien a los cambios. Peores para proyectos de desarrollo software de forma general. 	<ul style="list-style-type: none"> No funcionan muy bien con proyectos que necesitan de requisitos muy estrictos. Requiere de mayor control a lo largo del proyecto para la realización del seguimiento

Sin entrar profundamente en las diferentes metodologías que nos podemos encontrar ni utilizar actualmente, la anterior tabla ya nos da una idea de que no siempre las metodologías ágiles, particularmente SCRUM, son válidas para todas las situaciones. Ni tampoco SCRUM se aplica de una forma idéntica en todos los proyectos. La experiencia es uno de los valores más importantes por lo tanto para poder aplicar de una forma correcta una u otra metodología.



VÍDEO DE INTERÉS

Carlos Buenosvinos puede que sea uno de los mayores expertos en gestión de proyectos y en concreto sobre Agile. Un ejemplo sobre las reflexiones y el uso de Agile en este vídeo:



2.1 El manifiesto ágil

En el 2001, varios de los profesionales del software, ante las problemáticas de los modelos de gestión basados en procesos, se reúnen para debatir sobre otras posibilidades en metodología de proyectos.

En esa reunión surge el término **métodos ágiles**, como contraposición a las metodologías pesadas y rígidas. Surge también el manifiesto ágil, en el que se definen los valores sobre los que se asientan las metodologías.



ENLACE DE INTERÉS

El manifiesto ágil se encuentra en el siguiente enlace junto a los doce principios. Es muy interesante revisarlo.



Tal y como se indica en el enlace, allí se recogen los doce postulados del manifiesto, es un documento corto y conciso, pero que nos proporciona una idea general de los objetivos y la aplicabilidad.

Del manifiesto es importante destacar que ***se valora más a los individuos y su interacción que a los procesos y las herramientas***, es quizá uno de los postulados más importantes y sobre el que se basa la metodología Scrum. Es una revolución frente a las metodologías más tradicionales, donde el producto, el cómo, el cuándo y las especificaciones eran el centro de la gestión.

2.2 Cuando usar metodologías ágiles. Desarrollo de software

Ahora que ya tenemos un comparativo entre las metodologías ágiles y las no ágiles, así como un poco más de información sobre los principios de las metodologías ágiles, no podríamos plantear en qué momento deberíamos emplear las metodologías ágiles.

La respuesta no es totalmente cerrada, pero sí que podemos afirmar que en los escenarios o proyectos donde existen muchas variables, especificaciones o información que no se puede

cerrar, en esa tipología de proyectos podemos decir que las metodologías ágiles se comportan de una forma mucho mejor.



PARA SABER MÁS

Mediante este enlace podrás acceder a un interesante libro sobre la gestión de proyectos usando metodologías ágiles:



Dada esta afirmación entendemos que en los proyectos de desarrollo de software esté aplicándose tanto y con tan buen resultado. En el desarrollo de software es muy complejo poder cerrar todas las especificaciones, bien porque el cliente o el usuario no tiene conocimientos tecnológicos y por lo tanto no es capaz de transmitir todas las características, bien porque el negocio a implementar sea complejo o simplemente porque la realización de software mediante iteraciones provoca beneficios de cara a poder realizar modificaciones.

3. CONCEPTOS BÁSICOS SOBRE SCRUM

En el equipo de desarrollo se usan palabras como SCRUM, Scrum Máster y Sprint, conceptos que habías escuchado anteriormente, pero solo de forma teórica, no los habías puesto en práctica antes

Al comenzar a trabajar en el proyecto y con el equipo de trabajo, observas que se usan metodologías ágiles y te preguntas cuáles son los roles que se usan dentro del mismo. ¿Qué significado tiene la palabra Sprint y porqué le dan tanta importancia dentro del proyecto? ¿Cuál es el esquema general de un proyecto SCRUM?

Scrum es un modelo de desarrollo ágil. Scrum no nace a partir del manifiesto ágil como podríamos pensar, de hecho, la primera vez que se emplea una aproximación a esta metodología es dentro de sectores como la automoción. El término Scrum es la traducción de “melé” una formación típica del rugby, y por lo tanto estaba muy relacionada con el trabajo en equipo.

Esta metodología poco a poco fue evolucionando y aplicándose a otros sectores, entre ellos el desarrollo del software. Es cierto, que posteriormente a la creación del manifiesto ágil, se fue ajustando a lo que conocemos hoy en día como Scrum, y de hecho no fue hasta el 2009 con la creación de la Alianza Scrum y las certificaciones, donde se acaba de convertir en una metodología con mucho peso en el sector.



ENLACE DE INTERÉS

La web de la Alianza Scrum proporciona noticias, información y documentación sobre la metodología:

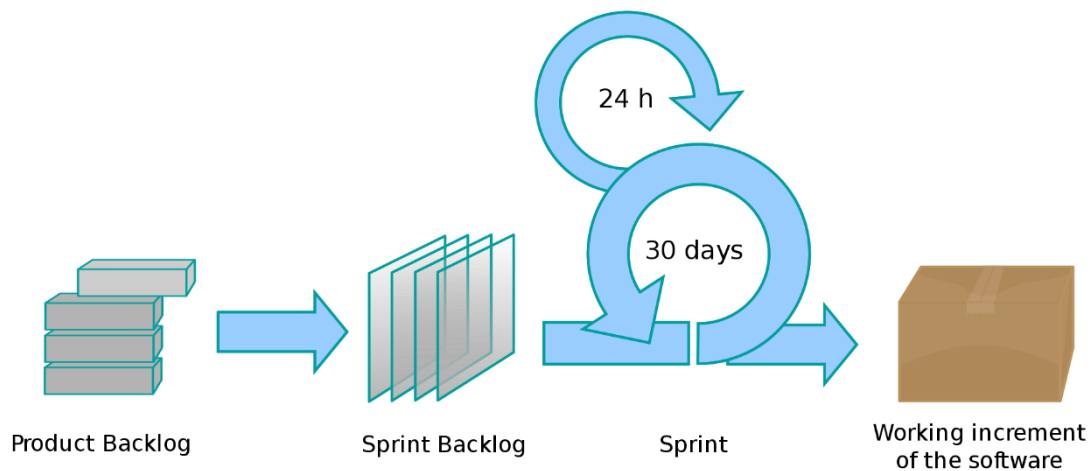


En pocas palabras, Scrum se caracteriza por:

- Ser una **metodología incremental**. Al contrario que las metodologías secuenciales o predictivas donde la planificación debe ser global.
- Tener **equipos auto organizados** que posean la responsabilidad y el conocimiento de los procesos implicados.

- **Procesos en paralelo** solapados en cada una de las iteraciones del proyecto, en vez de tareas consecutivas dependientes de su inicio y finalización unas de otras.

Una de las imágenes clave para la comprensión de la metodología Scrum es la que podemos ver a continuación:



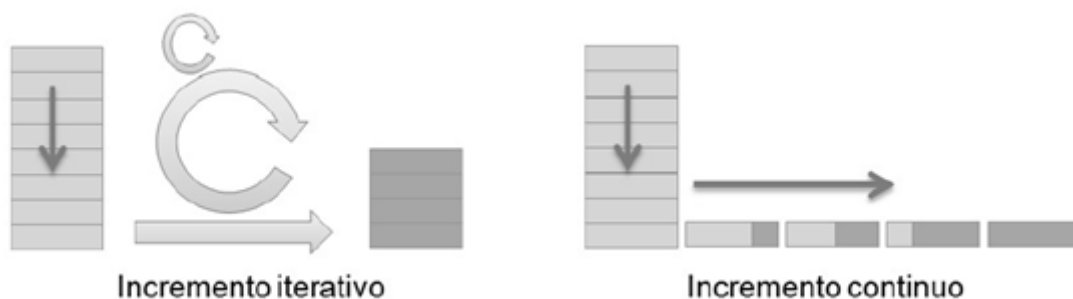
Marco Scrum.

Fuente: [https://en.wikipedia.org/wiki/Scrum_\(software_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development))

Aunque lo analizaremos posteriormente en detalle podemos ya observar:

1. Los términos usados están en inglés, sin traducción en muchos casos al castellano, como el término Sprint.
2. La importancia de las iteraciones para conseguir los objetivos marcados.
3. Lo poco estructurado que están los requisitos del desarrollo.

Si realizamos una comparación con una gestión continua, veremos claramente la diferencia entre una y otra metodología:



Marco iterativo vs continuo.

Fuente: Guía de formación Scrum Manager

3.1 Scrum técnico

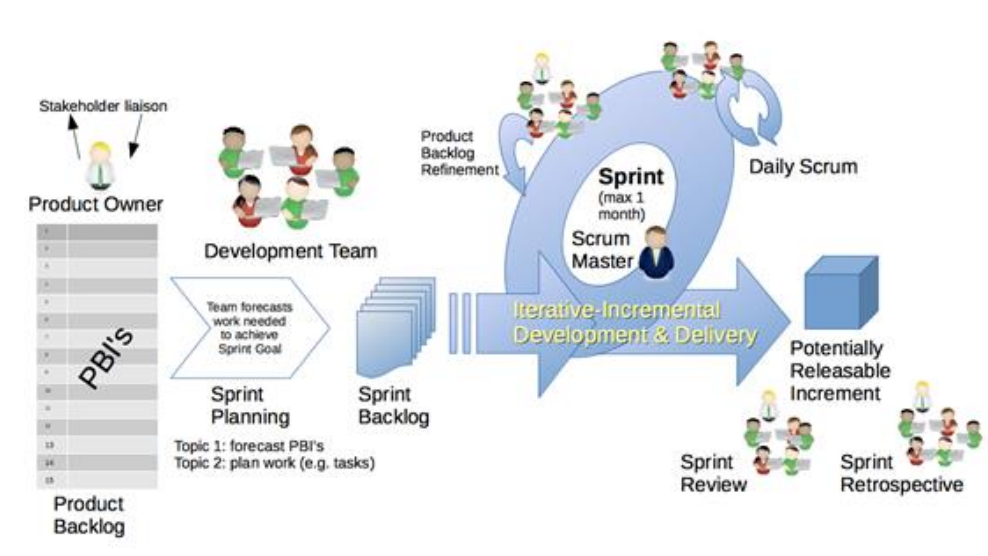
Llamamos Scrum técnico al conjunto de reglas y prácticas que se adoptan de acuerdo a los principios del desarrollo ágil:

- Una gestión evolutiva o iterativa del producto tal y como hemos visto en los anteriores diagramas.
- Autogestión de los equipos.

En concreto dentro del marco técnico resumido nos encontramos con los siguientes ámbitos y definiciones:

- Roles
 - El equipo Scrum.
 - El propietario del producto.
 - El Scrum Máster: es el responsable de velar por las normas de las diferentes fases dentro del proyecto.
- Los Artefactos
 - Pila de producto.
 - Pila del sprint.
 - El incremento.
- Los eventos
 - Sprint.
 - Reunión de planificación del sprint.
 - Scrum diario.
 - Revisión y retrospectiva del sprint.

De una forma más visual:



Roles, artefactos y eventos de Scrum.

Fuente: [https://en.wikipedia.org/wiki/Scrum_\(software_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development))

3.2 Artefactos

Denominamos artefactos a los elementos que nos permiten registrar la información, así como llevar un seguimiento de esta información a través del proceso. Definimos tres artefactos básicos:

- Pila de producto, lo podemos definir como la lista de requisitos de los diferentes usuarios que usan o usaran la aplicación. Uno de esos requisitos podría ser: “Como usuario, requiere el logueado en la aplicación usando el correo electrónico y contraseña de más de 7 caracteres”.
- Pila del sprint, sería la particularización de la lista de requisitos para el sprint en concreto.
- Incremento, el resultado producido en cada sprint o iteración.

Podríamos decir, que mediante algún mecanismo recogemos las necesidades de los diferentes usuarios que colocamos en la **pila de producto**, para ir agrupando de todos esos requisitos, los necesarios para **realizar un sprint** produciendo un producto que se denomina **incremento**.

3.3 Eventos

Podemos definirlos como diferentes estados o momentos dentro de la gestión mediante Scrum. Destacamos:

- Sprint, es cada una de las iteraciones que se producen en el desarrollo del producto. Es el eje principal de la metodología Scrum.
- Reunión de planificación del sprint, consiste en una reunión de trabajo al principio de cada sprint para poner en conocimiento y organizar las tareas de esa pila del sprint.
- Scrum diario, consiste en una breve reunión para coordinar el trabajo realizado, el previsto y las posibles necesidades o divergencias.
- Revisión del sprint, reunión que se produce al final del sprint para comprobar el incremento producido y las posibles adecuaciones y/o correcciones si se deben aplicar al siguiente sprint.

Una de las grandes diferencias con las metodologías más tradicionales, es la participación de todo el equipo de desarrollo, e incluso del propietario del producto al inicio y finalización del sprint.

3.4 Roles

Nos referimos a todas las personas que intervienen en el proyecto, ya porque trabajen de una forma directa, o porque indirectamente sean importantes para el producto. Destacamos:

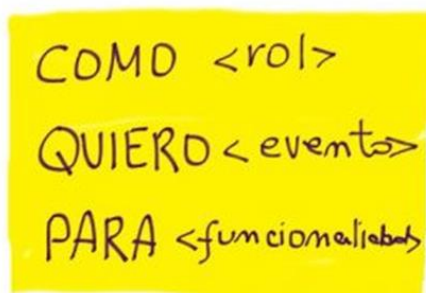
- Propietario del producto, consiste en la persona responsable de que el desarrollo del producto se adecue a las necesidades de los clientes y usuarios.

- Equipo de desarrollo, consiste en el conjunto de personas que se encargan de llevar a cabo el desarrollo del proyecto.

Scrum Máster, es el responsable de velar por las normas de las diferentes fases dentro del proyecto.

3.5 Historias de usuario

Dentro de la pila de producto se recoge la lista de requisitos, es lo que en metodologías más tradicionales entenderíamos como requisitos, sin embargo, dentro de SCRUM entendemos que debemos recoger la información siempre teniendo en nuestro punto de vista al usuario. De ahí nace lo que se denomina las Historias de Usuario, y que viene perfectamente representado en la siguiente imagen.



Historia de usuario.

Fuente: <https://urtanta.com/historias-de-usuario/>

Como vemos debemos traducir cualquier requisito en un: “Como usuario, quiero poder ver todos los cambios de productos en el almacén para poder pedir en tiempo y que no se produzca una rotura de stock.”



EJEMPLO PRÁCTICO

El jefe de departamento te encarga una primera tarea sencilla que es la de “traducir” parte de la conversación con el cliente en historias de usuario para el login. En concreto parte de la conversación con el gerente de la sala de congresos es la siguiente:

“Nos gustaría poder fidelizar a los clientes, conocer su nombre, edad y sexo es muy importante. Queremos que se registren en la aplicación y poder después saber quiénes son, teniendo en cuenta que el usuario debe realizar el menor número de pasos posibles.”

Para llevar a cabo la tarea debes centrarte en el login del usuario teniendo, por tanto, en cuenta a este usuario como centro de las historias de usuario. No realizarás las historias de usuario del tratamiento de esa información, ya que debes comenzar por esa parte sencilla. Dos posibles historias

que se pueden extraer son:

- Como gerente quiero que el usuario se registre en la aplicación incluyendo su nombre, edad y sexo para poder posteriormente realizar búsquedas y filtrados.
- Como usuario quiero poder registrarme en la aplicación usando el mínimo número de pasos para poder acceder a la aplicación.

4. CONCEPTOS BÁSICOS SOBRE KANBAN

Una vez que ya estás familiarizado con todos los conceptos sobre Scrum, como desarrollador que forma parte de un equipo de trabajo, ya te ves capaz de comenzar a participar en la gestión como Scrum team.

Para ello el responsable del departamento te incorpora al nuevo proyecto y te plantea las tareas que vas a asumir como responsable en el actual sprint y la importancia de cumplir con la estimación planteada. Pero ¿qué dependencias con otras tareas se plantean?

Como nos podremos imaginar, los conceptos y la metodología Scrum tiene una profundidad que no podemos abarcar en unas pocas páginas. Recordemos que existen certificados de varios niveles Scrum que acreditan el conocimiento y las capacidades para los futuros Scrum Master y gestores de proyectos.

Pero ¿Todos los proyectos se gestionan de una forma tan estricta? ¿En todos los proyectos se usan todos los artefactos, eventos y roles? La respuesta no es absoluta y general para todos los proyectos, de nuevo. Cada empresa, cada equipo y cada producto es diferente. Conocer la terminología y conocer los mecanismos más importantes es necesario para poder comenzar a trabajar o gestionar proyectos mediante metodología Scrum.

Una metodología que funciona muy muy bien con Scrum es KANBAN, que nos permite poder representar de una forma muy visual la pila de producto, tanto inicialmente como durante los sprints. Y una herramienta que representa perfectamente la metodología KANBAN y su aplicación dentro de SCRUM es Trello que veremos a continuación.

Kanban proviene de una palabra japonesa y que podríamos traducir como tablero visual. Igual que ocurrió con Scrum, sus orígenes se remontan al sector automovilístico y en concreto dentro del sistema de producción dentro de la marca Toyota.

Mientras que SCRUM es una metodología completa de planificación, de desarrollo, de seguimiento y control de la calidad. KANBAN nos encontramos que es un método que nos permite de una forma muy visual representar las diferentes tareas a realizar haciendo que el flujo de trabajo sea mucho más ágil.



Tablero KANBAN.

Fuente: <https://revistadigital.inesem.es/gestion-empresarial/kanban-el-metodo-para-desarrollar-proyectos-de-exito/>

Como dice la frase, una imagen transmite mucha más información que mil palabras, y la anterior imagen nos proporciona la base de la metodología KANBAN. Como podemos observar, el tablero KANBAN se representa con columnas donde se relacionan las diferentes tareas a realizar en al menos tres columnas: por hacer, haciendo, hechas. Así de sencillo podríamos resumir la metodología KANBAN.

Los objetivos de esta metodología son básicos pero muy importantes dentro de cualquier gestión de proyectos:

- Visualizar el trabajo, lo cual queda muy patente, ya que mediante estos tableros se puede tener una visión general de todo el escenario y del proyecto
- Gestionar las tareas ya que, mediante esta metodología, las tareas se van moviendo a lo largo de las columnas de una forma muy visual.
- Medir el tiempo, tanto la estimación como el tiempo real.

4.1 KANBAN y SCRUM

Las cuatro bases en las que se basa KANBAN son:

- Se puede usar e implementar en cualquier momento del proyecto.
- Permite adaptarse al cambio y evolución del equipo.
- No impone roles ni cargos, por lo que se puede acoplar perfectamente con otras metodologías.
- Alienta la autogestión.

La aplicabilidad de KANBAN dentro de Scrum es por lo tanto muy interesante. KANBAN puede ser usado para recoger la pila de producto, sus historias de usuario al comienzo de un proyecto. Una vez que se disponen de esas tareas, se puede pasar a una estimación del tiempo necesario para la realización de cada una de las tareas. Y por último la asignación de las tareas

al equipo Scrum también cuadra enormemente, porque permite la metodología una autogestión de las tareas por parte del equipo de una forma muy visual.

4.2 Trello

La aplicación de Kanban puede realizarse de una forma física, a través de notas (post-its) colocadas en una pizarra, ventana o pared. De hecho, veremos muchos centros de desarrollo de software con estos espacios con esta imagen.

También encontraremos herramientas para poder gestionar estos tableros, crearlos, compartirlos y usarlos. Y dentro de estas herramientas nos encontramos con Trello, como la herramienta más conocida para la creación de tableros Kanban.



ARTÍCULO DE INTERÉS

Aquí puedes leer un interesante artículo sobre cómo comenzar a usar Trello:



4.2.1 Comenzando a usar Trello

Trello es de las denominadas Software As a Service, es decir, la aplicación no es necesario tener que instalarla y sencillamente tendremos que registrarnos para comenzar con un nuevo tablero tal y como vemos a continuación.



VÍDEO DE INTERÉS

Aquí podrás ver paso a paso el registro e inicio de uso de Trello:



4.2.2 Primer tablero y tareas

Una de las características más importantes es la facilidad de uso de la aplicación, cómo en pocos pasos podemos crear un nuevo tablero, y cómo también de forma muy intuitiva podemos comenzar a crear nuevas tareas tal y como vemos a continuación.



VÍDEO DE INTERÉS

En este video se muestran las indicaciones para crear un nuevo tablero y una introducción a la creación de tareas en Trello:



4.2.3 Gestión colaborativa

La gestión de proyectos dentro de equipos heterogéneos, deslocalizados y autogestionados es otra de las grandes características que nos encontramos cuando trabajamos con Trello tal y como vemos en el siguiente tutorial.



VÍDEO DE INTERÉS

En este vídeo configuraremos y compartiremos un tablero con otros usuarios:



EJEMPLO PRÁCTICO

En la aplicación sobre recogida de la afluencia de eventos, nuestro jefe nos propone realizar las tareas dentro del tablero creado y compartido con nosotros para la realización del interfaz que se conectará a la bbdd ya creada. En concreto, la interfaz debe reflejar el registro y/o login del usuario, introduciendo en el caso del registro, usuario, contraseña, nombre y apellidos, edad y sexo.

El paso a paso de la resolución lo podemos visualizar de una forma más detallada en este vídeo:



RESUMEN FINAL

La gestión de proyectos es una de las partes más importantes dentro de un equipo de desarrollo dentro de cualquier empresa. El éxito o el fracaso de una aplicación depende en muchas ocasiones de esa buena planificación y seguimiento.

A lo largo de la unidad hemos visto como la gestión ha ido evolucionando, de forma paralela también a la evolución del entorno. Y dentro de esa evolución, las metodologías ágiles se adecúan justamente a los nuevos equipos de trabajo, de clientes y del ciclo de desarrollo.

Una de las metodologías ágiles donde hemos profundizado es SCRUM, donde las personas pasan a ser el centro del proceso, un proceso iterativo donde los equipos heterogéneos y auto gestionados son el centro justamente de esta metodología cada vez más usada.

Para finalizar hemos introducido KANBAN, como otra metodología que perfectamente puede usarse junto a SCRUM.