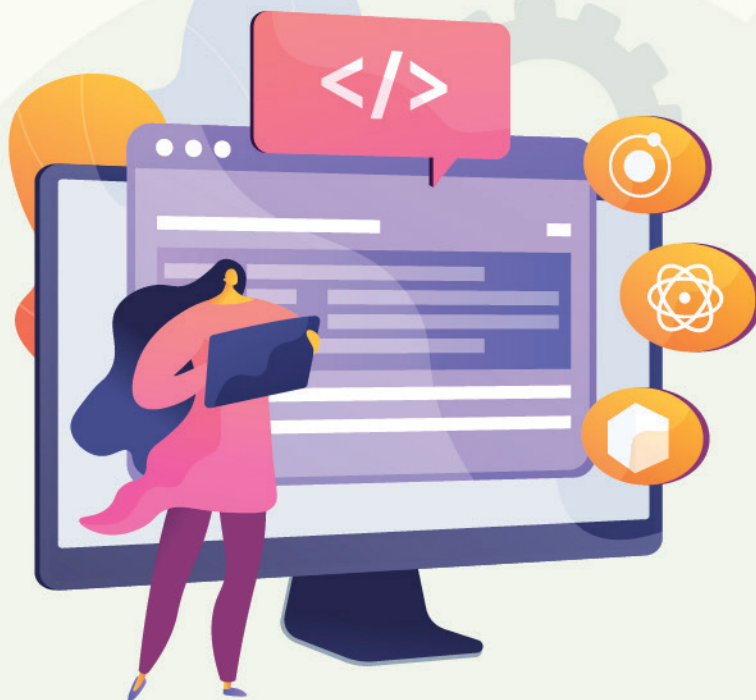




CICLO 3

[FORMACIÓN POR CICLOS]

Metodología **SCRUM**



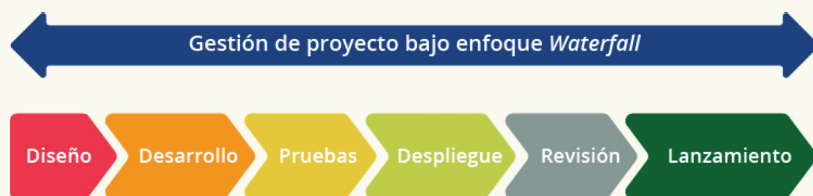
Ingeni@
Soluciones TIC



**UNIVERSIDAD
DE ANTIOQUIA**
Facultad de Ingeniería

Los proyectos de desarrollo de *software* pueden ser gestionados utilizando dos tipos de enfoques: *waterfall* o *agile*.

Waterfall, o *cascada*, era hace unos años el enfoque más utilizado para desarrollar productos de *software* y se caracteriza por ser un esquema de trabajo secuencial, controlado y estricto. Bajo este enfoque el proyecto se divide en etapas tales como captura de requisitos, diseño, desarrollo, *testing*, corrección de errores e implementación. Cada una de estas fases es independiente, y cuando una de ellas termina, la siguiente comienza, por lo cual transcurre un tiempo considerable hasta que los clientes reciben resultados. En un proyecto gestionado bajo un esquema *waterfall* hay margen escaso para mejorar y solucionar problemas porque la retroalimentación y la validación del cliente se reciben una vez finalizada la fase de desarrollo.

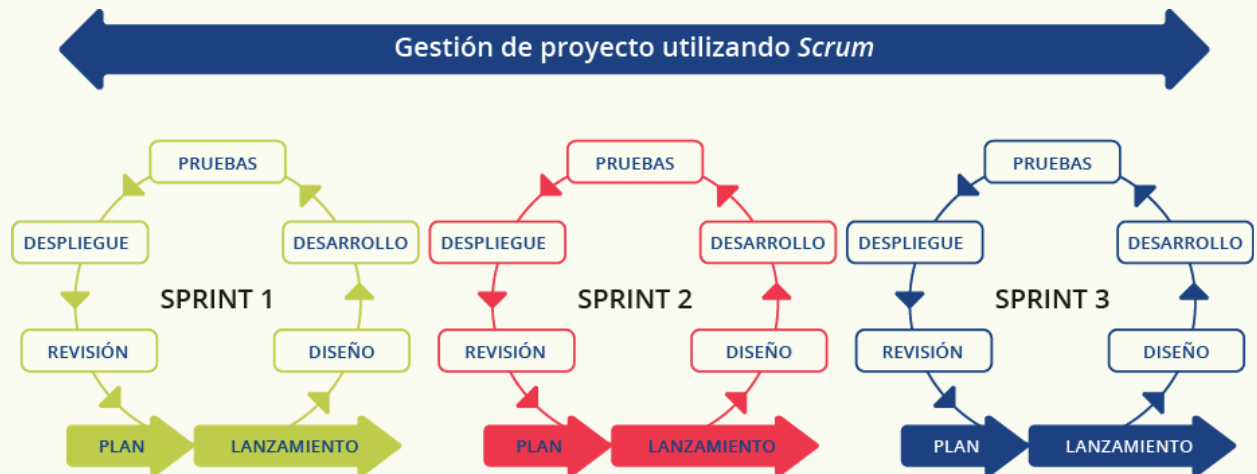


Por otro lado, las metodologías ágiles, o *agile*, son un enfoque para gestionar los proyectos centrándose en los cambios y la colaboración, e implican una forma distinta de trabajar y de organizar el equipo. Este enfoque busca la mejora continua fomentando entregas frecuentes y periódicas al cliente desde el inicio del proyecto con el fin de entregar valor, hacer validaciones y recibir retroalimentación de manera constante. Entre sus ventajas se encuentran la flexibilidad, el enfoque en el usuario, el aumento de calidad del producto, la revisión a tiempo para hacer correcciones y ajustes, el aumento en la productividad por la realización de trabajo en forma simultánea y las entregas en ciclos cortos. Las metodologías ágiles requieren un alto nivel de involucramiento del cliente durante toda la gestión del proyecto.

Específicamente, **scrum** es una metodología ágil de gestión de proyectos, que se basa en el desarrollo de entregas iterativas y en una planificación con adaptación a los cambios y el progreso del proyecto.



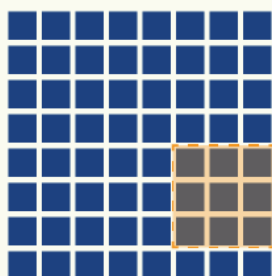
Dentro del marco de trabajo de **scrum**, el proyecto se ejecuta dividiéndolo en ciclos de tiempo cortos de duración fija, denominados *sprints*. Cada *sprint* es una iteración en la que se desarrolla una parte del producto e incluye las etapas de planeación, diseño, desarrollo, pruebas, despliegue, revisión y lanzamiento.



El proceso parte de un levantamiento de las necesidades y deseos del cliente, denominado *product backlog*, que se documenta en forma de *historias de usuario* al inicio del proyecto. El product backlog es dividido en bloques de trabajo que se desarrollan en cada uno de los *sprints*. Al final de cada *sprint* se entrega al cliente una parte del producto final, generando entregas incrementales que conforman el resultado final del proyecto. La duración de cada *sprint* oscila entre una y cuatro semanas.

Product Backlog


Lista de necesidades y requerimientos del cliente escritas en forma de **historias de usuario**.



Product Backlog

Conjunto de historias de usuarios que se desarrollarán durante el sprint.





Al inicio de cada *sprint*, el equipo *scrum* realiza una reunión de planeación en la que se selecciona el conjunto de historias de usuario a desarrollar y se decide cómo complementarlas. De igual manera, durante cada *sprint* se realizan reuniones diarias denominadas *daily*, en las que, en máximo 30 minutos, el equipo discute el progreso, los pasos a seguir, las inquietudes y los riesgos sobre las tareas que cada miembro está desarrollando.

Historias de usuario

Como se mencionó anteriormente, los requerimientos del producto, o *product backlog*, consisten en el listado de todas las tareas que se pretenden hacer durante el desarrollo de un proyecto y se escriben en forma de *historias de usuario*. El *product backlog* debe ser visible ante todo el equipo del proyecto con el fin de que este cuente con una visión panorámica de todas las actividades que se esperan realizar.

Las *historias de usuario* están pensadas para ayudar al equipo de proyecto a centrarse en el cliente. Por esta razón, su redacción se realiza desde el punto de vista de los diferentes roles de usuario.

En la siguiente imagen se muestra la estructura de una historia de usuario. Su redacción debe iniciar con la palabra “como” seguida por el nombre del rol a la que está dirigida la historia de usuario. Luego se escribe la palabra “necesito” seguida de la descripción de la funcionalidad.

Estructura de una historia de usuario

Como	[Nombre del rol]
Quiero	[Descripción de la funcionalidad]



Ejemplo

Un equipo *scrum* está desarrollando una plataforma de gestión de contratación de personal a una empresa del sector industrial.

El *product owner* del proyecto será el líder de contratación del área de gestión humana de la empresa. Esta persona también será el administrador de la plataforma una vez comience la operación de la misma.

La plataforma consiste en una página web que permite que un administrador y un candidato a un cargo interactúen y tengan trazabilidad del proceso de contratación.

A partir de los requerimientos y necesidades de cada usuario, el *scrum master* lidera una reunión en la que se hace el levantamiento de las siguientes historias de usuario para el desarrollo de la plataforma.

Item	Rol	Descripción	Historia de usuario
HU-1	Administrador	Crear una nueva vacante. Para crear una vacante, debo poder indicar el cargo de la vacante, la cantidad de candidatos necesarios para la vacante, la descripción del cargo, el rango del salario y la fecha planeada de inicio del trabajo	Como administrador, quiero poder crear una nueva vacante. Para crear una vacante debo poder indicar el cargo de la vacante, la cantidad de candidatos necesarios para la vacante, la descripción del cargo, el rango del salario y la fecha planeada de inicio del trabajo
HU-2	Administrador	Crear un nuevo candidato para un proceso de contratación. Para el candidato debo poder agregar el nombre, la cédula, el correo electrónico personal y el teléfono. Adicionalmente, debo poder indicar un usuario temporal para el candidato, con el cual podrá ingresar al sistema. Dicho usuario tendrá el formato de un correo electrónico. Una vez se cree el usuario para el candidato, el sistema generará una contraseña temporal que el administrador deberá compartir al candidato	Como administrador, quiero poder crear un nuevo candidato para un proceso de contratación. Para el candidato debo poder agregar el nombre, la cédula, el correo electrónico personal y el teléfono. Adicionalmente, debo poder indicar un usuario temporal para el candidato, con el cual podrá ingresar al sistema. Dicho usuario tendrá el formato de un correo electrónico. Una vez se cree el usuario para el candidato, el sistema generará una contraseña temporal que el administrador deberá compartir al candidato
HU-3	Administrador	Crear un nuevo evento de entrevista, en el cual podré registrar la vacante, el candidato, la fecha y hora del evento y los detalles de la ubicación o el <i>link</i> de la reunión	Como administrador, quiero poder crear un nuevo evento de entrevista, en el cual podré registrar la vacante, el candidato, la fecha y la hora del evento y los detalles de la ubicación o el <i>link</i> de la reunión

Item	Rol	Descripción	Historia de usuario
HU-4	Candidato	Iniciar sesión con el correo temporal que me entregó el administrador	Como candidato, quiero poder iniciar sesión con el correo temporal que me entregó el administrador
HU-5	Candidato	Cambiar mi información de perfil, como mi teléfono, mi identificación, mi foto o mi dirección	Como candidato, quiero poder cambiar mi información de perfil, como mi teléfono, mi identificación, mi foto o mi dirección
HU-6	Candidato	Visualizar las entrevistas que tengo programadas en el sistema	Como candidato, quiero poder visualizar las entrevistas que tengo programadas en el sistema

Beneficios

La metodología *scrum* se centra en que las personas hagan el trabajo con eficacia, y en la comunicación y retroalimentación constante por parte del cliente para desarrollar un producto que entregue la mayor cantidad de valor posible. Adicionalmente, tiene un enfoque iterativo e incremental que permite controlar el riesgo, aprender y adaptarse a las necesidades de los usuarios.

Los principales beneficios de la metodología se muestran a continuación:

- Entrega de valor constante y desde el inicio del proyecto
- Desarrollo iterativo e incremental
- Mejora continua en la gestión del proyecto y del equipo de trabajo
- Diseño centrado en el usuario
- Testeo constante para permitir identificar errores y riesgos a tiempo
- Seguimiento, retroalimentación y acción en ciclos cortos

Actores dentro del equipo scrum

El equipo *scrum* está conformado por las personas encargadas del desarrollo y de todas las actividades relacionadas con el producto, desde la colaboración con los *stakeholders*, el testeo, el diseño y el desarrollo. Todo el equipo es responsable de entregar un resultado en cada

sprint para construir un producto que crece de manera incremental a medida que avanza el proyecto.

El equipo *scrum* es un equipo autogestionado, en el que cada miembro decide qué hace y cómo lo hace bajo la guía del *scrum master*.

Un equipo *scrum* está conformado por el *product owner*, el *scrum master*, los desarrolladores y los *stakeholders*.



Product owner

El *product owner* representa la voz del cliente dentro del equipo *scrum*. Se encarga de construir y revisar el *product backlog* y priorizar las historias de usuario.

Es la persona encargada de transmitir los requerimientos del cliente al equipo *scrum*, asegurando que el equipo tenga claro el objetivo del producto y priorizando las tareas para que el equipo entregue resultados que agreguen valor al cliente. Adicionalmente, se encarga de revisar y proporcionar retroalimentación sobre el progreso del producto.

El *product owner* asegura que el *product backlog* sea transparente, visible y entendible para todos los miembros del equipo. Por esta razón, su involucramiento y compromiso es fundamental para el éxito del proyecto.



Scrum master

El *scrum master* es el responsable de asegurar que la metodología *scrum* se implemente de manera adecuada durante el desarrollo del proyecto.

Se encarga de capacitar y guiar al equipo en los principios de la metodología *scrum*, ayudando a eliminar las barreras que se pueden presentar para el desarrollo del proyecto.

El *scrum master* es el líder del equipo *scrum*, y organiza la gestión de tareas para que se realicen de manera eficiente.

OJO: ¿Incluimos algo de seguimiento y control de cronograma, aprobaciones, decisiones?



Desarrolladores

Los desarrolladores son el grupo de personas encargadas de la ejecución y materialización del producto. Su responsabilidad es construir un entregable valioso y de calidad en cada *sprint*.



Stakeholders

Los *stakeholders* o interesados son las personas u organizaciones que afectan o se ven afectadas por el desarrollo del producto.

Los *stakeholders* pueden ser actores con poder de decisión, personas o grupos que participan en cierta parte del proceso con aprobaciones o tareas específicas.

Algunos ejemplos de *stakeholders* son el cliente, el departamento legal, el departamento de operaciones, los proveedores y las entidades regulatorias, entre otros.

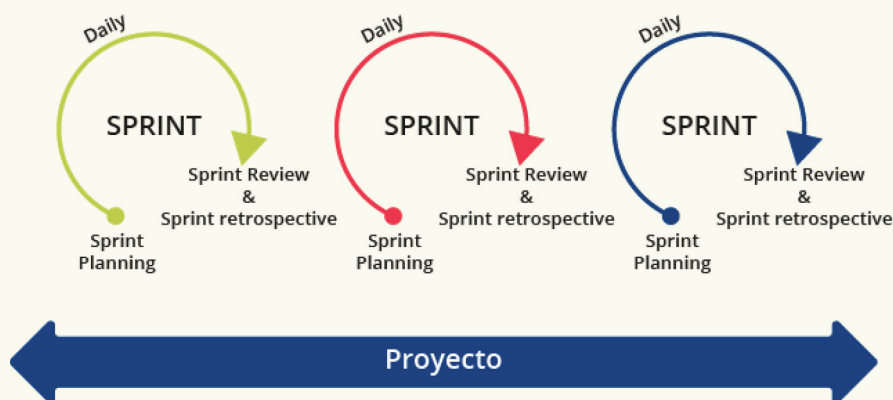
Eventos

La implementación de la metodología *scrum* requiere que el equipo lleve a cabo ciertos eventos y rutinas que permitirán organizar el trabajo y establecer rutinas para incrementar la eficiencia:

- **Sprint.** Los *sprints* son el corazón de la metodología *scrum* porque son los bloques de tiempo en los que el equipo entrega resultados al cliente. Son eventos de una duración fija, que oscilan entre una semana y un mes. Un *sprint* comienza inmediatamente después de la finalización del *sprint* anterior.

Durante un *sprint* se lleva a cabo una serie de eventos para organizar la ejecución del trabajo. Al inicio de cada *sprint* se realiza un *sprint planning* o reunión de planeación para definir las historias de usuarios que se harán durante el *sprint*. Durante todos los días hábiles del *sprint* se realiza un *daily* o reunión diaria para hacerle seguimiento al progreso de las actividades que está ejecutando el equipo. Finalmente, al culminar cada *sprint* se lleva a cabo un *sprint review* y un *sprint retrospective* para consolidar resultados antes de que inicie el siguiente *sprint*.

Eventos Scrum



• **Sprint planning.** Es una sesión de trabajo en la que se define el conjunto de historias de usuario que se llevarán a cabo durante el sprint. Esta sesión es liderada por el *scrum master* y se realiza al inicio de cada *sprint*. Durante la reunión el equipo debe entender por qué las historias de usuario que desarrollará son relevantes y cómo agregan valor al producto final. De igual manera, el equipo debe discutir cómo se realizarán las historias de usuario seleccionadas y cuánto tiempo tomará su ejecución.

Sprint Planning

1. ¿Por qué es valioso este sprint?

Definir el objetivo del sprint y su impacto en la construcción producto final.



2. ¿Qué historias de usuarios se ejecutarán durante este sprint?

A través de una conversación con el producto owner y la guía del scrum master, los desarrolladores definen los elementos del Product Backlog (Historias de Usuario) que ejecutarán durante el sprint.



3. ¿Cómo realizarlo?

Se asigna una carga de trabajo en número de horas para cada historia de usuario y desarrollador, se realizan preguntas y resuelven dudas, y se pule el requerimiento.

La *sprint planning* tiene un límite de tiempo de máximo ocho horas para un *sprint* de un mes. Para *sprints* más cortos, el evento suele ser de menor duración.

• **Daily.** Es una reunión de entre 10 y 30 minutos que se realiza todos los días a la misma hora y en el mismo lugar. A esta reunión asiste el equipo *scrum*, incluidos el *product owner* y el *scrum master*. Durante la sesión, cada uno de los desarrolladores comparte al equipo las actividades que completó, las actividades que están en proceso de desarrollo y las actividades que realizará durante el día. Adicionalmente, se mencionan inquietudes y posibles barreras o riesgos para la ejecución del trabajo. En caso de ser necesario, se programa una reunión adicional para solucionar temas puntuales.

El principal objetivo de esta reunión es mejorar la comunicación, identificar impedimentos y tomar decisiones de manera ágil, facilitando la ejecución de actividades.



• **Sprint review.** Es una reunión que se realiza al final de cada *sprint* y en la que cada desarrollador presenta los resultados de dicho *sprint* a los interesados clave y discute el progreso general de la ejecución del proyecto.

Al finalizar cada *sprint review* se ajusta y actualiza el *product backlog* según el avance que se obtuvo durante el *sprint*.

• **Sprint retrospective.** Es una reunión a la que asisten el *product owner*, el *scrum master* y los desarrolladores con el objetivo de definir acciones para aumentar la calidad y la efectividad del equipo.

Comienza con una discusión del desempeño de las personas, los procesos y las herramientas durante el *sprint*, identificando aquello que salió mal y cómo puede ser mejorado. Al final de la reunión, se definen acciones claras para mejorar el desempeño como equipo. Estas acciones pueden incluir cambios de herramientas, ajustes de procesos y dinámicas de equipo, y pueden ser incluidas en el *product backlog*.



Bibliografía

Para más información, consulte la guía de scrum:

<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf>

