

1. Organización de la práctica

Este documento contiene la información necesaria para realizar la primera parte de la práctica 1 de la asignatura. Esta práctica tiene una duración de tres semanas, en las que se explicarán los conceptos básicos relacionados con la gestión del proyecto software que se va a desarrollar durante el cuatrimestre. A continuación, se detallan los objetivos de esta primera parte de la práctica:

Gestión del proyecto con *Scrum* y *YouTrack*

- **Objetivos:**
 1. Conocer los fundamentos básicos de la metodología ágil *Scrum*.
 2. Comprender la importancia de la gestión de tareas en un proyecto software.
 3. Aprender a manejar la aplicación *YouTrack*.
- **Preparación:** Los estudiantes deben haber formado los grupos de trabajo (tres personas), y deberán haberse creado una cuenta gratuita en la plataforma *YouTrack*¹.
- **Seguimiento y evaluación:** tras explicar los conceptos teóricos básicos, los estudiantes crearán en *YouTrack* un proyecto para su grupo de trabajo. Será necesario crear un único proyecto en *YouTrack* por cada grupo, donde se deben añadir como miembros a todos los componentes del grupo, así como al profesor responsable del grupo de prácticas. Durante el resto de la sesión, se explicará el funcionamiento de la aplicación y los estudiantes irán probando sus funciones para familiarizarse con el entorno. La práctica no tiene evaluación, pero se comprobará que han realizado ejemplos sobre la plataforma.

¹https://www.jetbrains.com/es-es/youtrack/download/get_youtrack.html

2. Gestión del proyecto con *Scrum* y *YouTrack*

2.1. Introducción a la metodología *Scrum*

Para el desarrollo del proyecto de prácticas se seguirá una metodología ágil basada en *Scrum* [1]. *Scrum* es un marco de trabajo apropiado para el desarrollo y mantenimiento de un producto (no solo software), donde el eje principal es la entrega del producto con el máximo valor posible, siempre buscando la eficacia de las prácticas de gestión y desarrollo [1]. Para ello, se emplea un modelo iterativo e incremental centrado en los tres pilares fundamentales de un método empírico:

1. Transparencia. Todos los participantes deben tener un lenguaje común que les permita entender el proceso.
2. Inspección. La metodología se fundamenta en la revisión continua de los artefactos y del progreso hacia el objetivo final. Estas revisiones deben detectar posibles desvíos sin obstaculizar el trabajo diario.
3. Adaptación. Se deben determinar las causas de los desvíos y ajustar el proceso para minimizarlos y alcanzar un producto satisfactorio para el cliente.

Scrum está especialmente diseñado para el trabajo colaborativo en equipos, poniendo en relieve la importancia de organizarse de forma autónoma y aprender de la propia experiencia a medida que se avanza en el problema [2]. El compromiso, coraje, foco, apertura y respeto son los valores que *Scrum* fomenta en sus equipos. Para adoptar la metodología *Scrum* es necesario conocer sus tres componentes principales (roles, artefactos, y eventos), así como las reglas que rigen las relaciones e interacciones entre ellos. No obstante, la metodología deja abierta la elección de las estrategias concretas con las que poner en práctica estos conceptos.

En la Figura 1 se muestra una visión global de la metodología *Scrum* y sus componentes. El elemento central de *Scrum* es la iteración o *sprint*. Un *sprint* es un periodo breve de tiempo fijo durante el cual el equipo debe completar una parte del producto [2]. Dividiendo el trabajo en *sprint*, se busca facilitar su gestión y favorecer la adaptación a cambios. Al final de cada *sprint* se debe estar en disposición de entregar un producto que funciona, pese a que no esté aún completo.

scrum

Proceso de la metodología Ágil

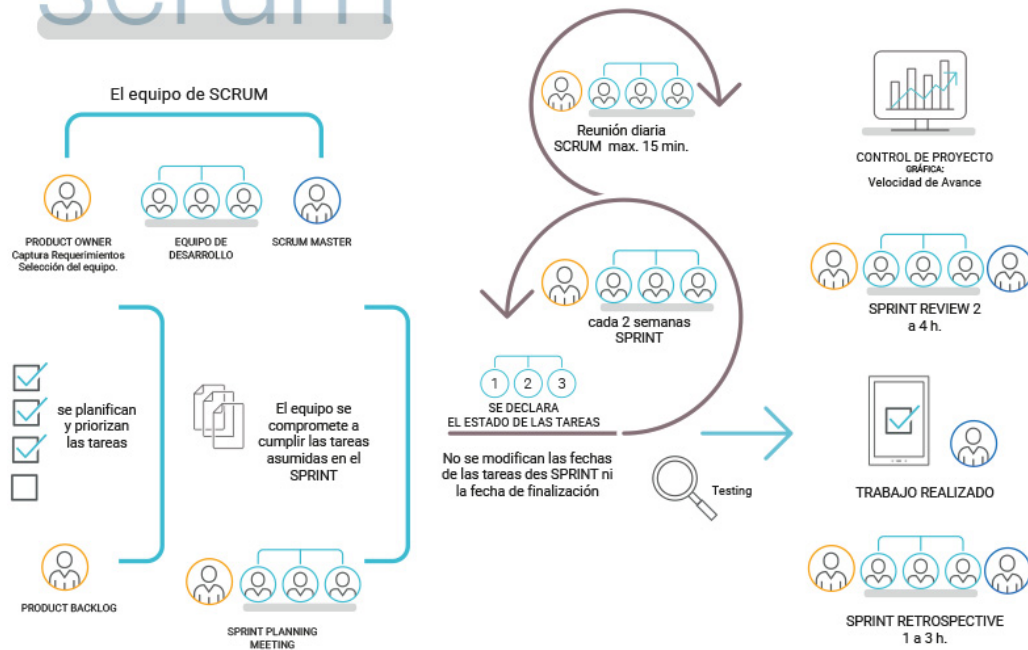


Figura 1: Metodología *Scrum*. Fuente: <https://development.grupogaratu.com/metodologia-scrum-desarrollo-software/>.

2.2. Roles, artefactos y eventos

En esta sección se describen los componentes principales de la metodología *Scrum*: roles, artefactos y eventos o reuniones. En la Figura 2 se muestra un resumen de dichos componentes.

Roles. Un equipo *Scrum* está formado por tres roles: el dueño del producto, el equipo de desarrollo y un gestor experto en la metodología.

- Dueño del producto (*product owner*). Es el máximo responsable del trabajo que va a desarrollar el equipo de trabajo. Debe establecer los objetivos a alcanzar y asegurar que todo el mundo los entiende. Se encarga de definir la lista de requisitos a desarrollar en el *sprint* (llamada “lista de producto”) y establecer prioridades.
- Equipo de desarrollo (*development team*). Se compone de un grupo de profesionales que se autoorganizan para realizar el trabajo asignado al *sprint*. Todos los miembros tienen el mismo grado de responsabilidad, aunque pueden tener

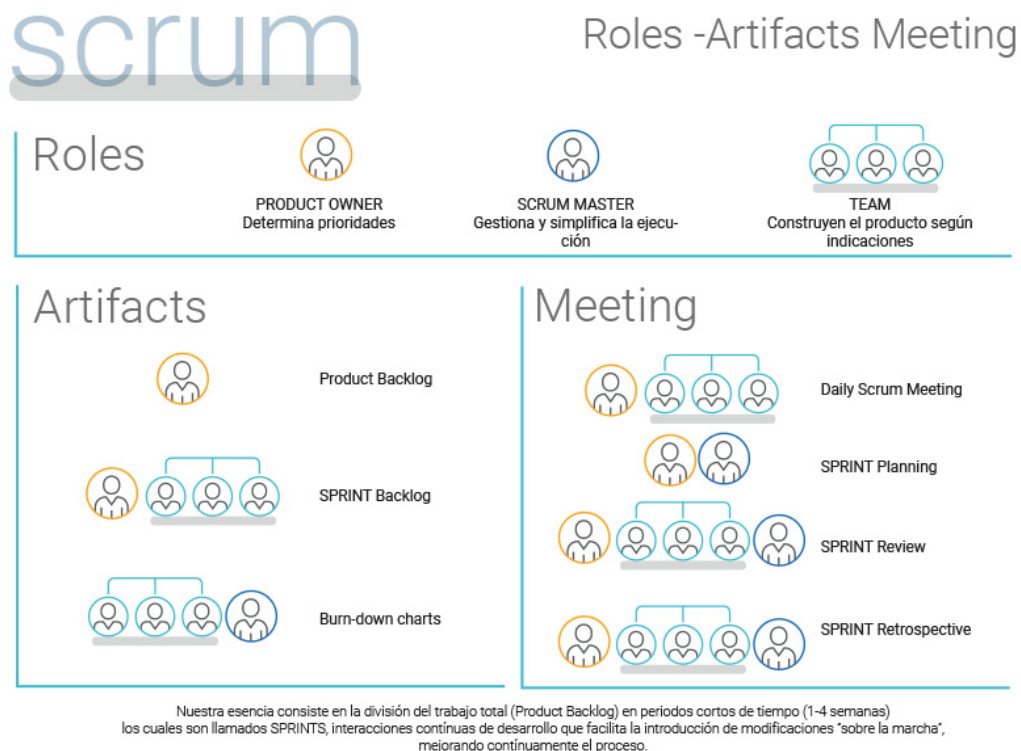


Figura 2: Roles, artefactos y eventos en *Scrum*. Fuente: <https://development.grupogaratu.com/metodologia-scrum-desarrollo-software/>.

habilidades especializadas que dotan al grupo de un carácter multidisciplinar. Su tamaño debe ser el adecuado para que el desarrollo sea ágil y se garantice la entrega. La guía de *Scrum* recomienda equipos de entre tres y nueve miembros [1], que pueden incluir personas con otros roles.

- Gestor experto en la metodología (*Scrum master*). Es el líder encargado de garantizar que el resto de personal entiende y adopta la metodología *Scrum*. Entre sus tareas destacan ayudar al *product owner* a crear la lista de producto para maximizar el valor del mismo, guiar al equipo de desarrollo, facilitar la organización de las reuniones que se requieran y, en general, apoyar a la organización a optimizar la puesta en práctica de la metodología.

Artefactos. Son documentos de trabajo en formatos diversos que dan soporte al proceso *Scrum*, permitiendo la trazabilidad y transparencia de la información clave.

- Lista de producto (*Product backlog*). Es una lista ordenada de los requisitos del producto. Su responsable es el dueño del producto, que debe garantizar su corrección y disponibilidad. Esta lista evoluciona a medida que se va desarrollando el producto, detallando todas las características, funcionalidades,

mejoras y correcciones que se van realizando. Cada elemento de la lista lleva asociado: una descripción, un orden, una estimación de tiempo, y un valor.

- Lista de “pendientes” (*Sprint backlog*). Es el subconjunto de elementos de la lista de producto que se consideran en el *sprint* actual, junto a su plan de desarrollo. Debe tener el suficiente nivel de detalle para que sirva de guía a los miembros del equipo de desarrollo, quienes pueden añadir o eliminar elementos a medida que se completa el *sprint*. Permite conocer en todo momento las tareas pendientes, en desarrollo o terminadas.
- Gráficas de seguimiento (*Burn-down charts*). Aunque no se describen oficialmente en la guía de *Scrum*, es habitual generar gráficas de seguimiento para evaluar el cumplimiento de objetivos y la productividad del equipo.

Eventos. Una de las singularidades de *Scrum* es la definición y organización de eventos (bloques de tiempo con duración máxima establecida) que optimicen el aprovechamiento del tiempo fomentando así la productividad. Son necesarios para garantizar la transparencia e inspección, ya que en ellos se definen las tareas a realizar y se revisa el trabajo concluido. Todos los eventos giran en torno al *sprint*.

- Reunión diaria (*daily Scrum meeting*). Es una reunión corta, de 15 minutos, y que se recomienda que sea de pie para evitar su prolongación. En ella, el equipo de desarrollo distribuye las tareas a realizar en ese día y evaluar si el progreso es el adecuado.
- Planificación (*Sprint planning*). Se establece la funcionalidad que va a incorporarse al producto en el siguiente *sprint* y se decide cómo se realizará el trabajo para lograr dicho objetivo. Aunque el peso de esta actividad recae en el *product owner*, quién habrá priorizado la lista de producto, es importante que también participe el equipo de desarrollo para aportar su visión sobre el desarrollo y garantizar que todos comprendan el objetivo.
- Revisión (*Sprint review*). Al terminar el *sprint*, todo el equipo analiza el trabajo realizado junto al cliente. Se trata de una reunión informal donde se informa del estado actual, se demuestra el funcionamiento del producto y se recopila información que sea de utilidad para planificar los siguientes *sprints*.
- Retrospectiva (*Sprint retrospective*). Se realiza tras la revisión del *sprint* y su máximo responsable es el *scrum master*. Su finalidad es analizar el *sprint* respecto a las personas, procesos, etc., con el fin de identificar problemas y plantear mejoras.

2.3. Plataforma *YouTrack*

YouTrack es una plataforma web, también disponible como aplicación de escritorio, para la gestión de proyectos. Sus principales funcionalidades son:

- Gestión de incidencias. Permite crear y hacer el seguimiento de las incidencias que surgen durante el desarrollo del proyecto.
- Base de conocimiento. Es un área donde se pueden crear artículos, notas sobre reuniones, documentación, etc. Soporta lenguaje *Markdown*.
- Generación de informes. Dispone de una gran variedad de gráficos e informes para analizar el progreso del proyecto y cómo se distribuye el trabajo entre los miembros del equipo.
- Gestión del tiempo. Permite incorporar estimaciones a las tareas y realizar su seguimiento, así como crear diagramas de Gannt para la planificación global.

Además, *YouTrack* permite la creación de “paneles ágiles” inspirados en el concepto de *sprint*. Aunque no implementa los elementos propios de *Scrum*, la plataforma ofrece la posibilidad de crear tareas en cada *sprint* y asignarles un responsable, una prioridad y una estimación de tiempo. Las tareas pueden estar en cuatro estados: abiertas, en progreso, por verificar, y terminadas. Por defecto, cada proyecto en *YouTrack* tiene dos paneles: uno para la gestión del proyecto (*project management*) y otro para el desarrollo (*project development*).

Referencias

- [1] Ken Schwaber and Jeff Sutherland. La guía de scrum. *Scrumguides. Org*, 1:21, 2013. Disponible en: <https://scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Spanish.pdf>.
- [2] Atlassian. Agile coach. Disponible en: <https://www.atlassian.com/es/agile/scrum>.