Contenidos

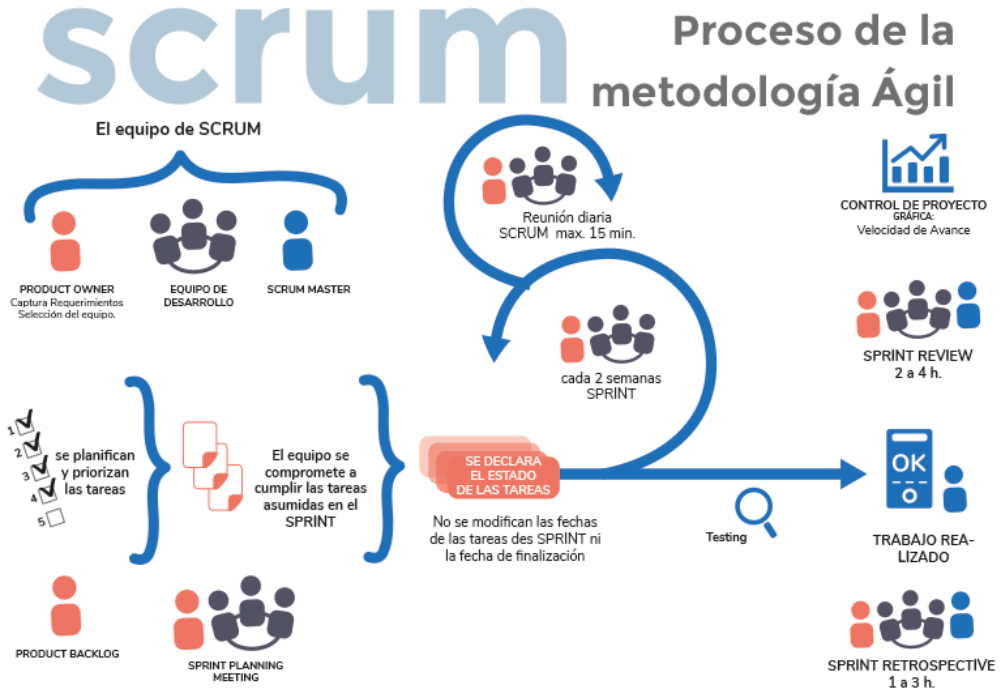
0. Desarrollo de las clases

0.1. SCRUM

SCRUM

- Es una metodología ágil que se basa en entregas parciales y frecuentes de un producto para obtener resultados con rapidez.
- Es una metodología incremental, cuyos objetivos varían debido a que están poco definidos.

SCRUM



SCRUM

Principios de SCRUM:

- **No desperdiciar** el tiempo.
- La **calidad** del producto comienza en el principio del desarrollo.
- **Crear conocimiento**.
- Tomar las decisiones en el **momento idóneo**.
- Entregas **más rápidas**.
- **Motivar** a todo el equipo.
- **Optimizar** todos los procesos al máximo.

SCRUM

Tiene los siguientes roles:

- **Product Owner**: marca las pautas de actuación del proyecto general.
- **Scrum Master**: guía las reuniones y coordina el equipo.
- **Team**: implementan las funcionalidades.
- **Users**: beneficiarios finales.

SCRUM

Tiene las siguientes herramientas:

- **Product Backlog**: contiene todas las funcionalidades ordenadas por prioridades. Es la agrupación de todas las historias de usuario del producto.
- **Sprint Backlog**: funcionalidad a desarrollar en un *sprint* determinado. Se crea durante el *sprint planning meeting*.
- **Burndown Chart**: gráfico que muestra cantidad de trabajo hecho.

SCRUM

Product backlog. Estructura de cada elemento (historia):

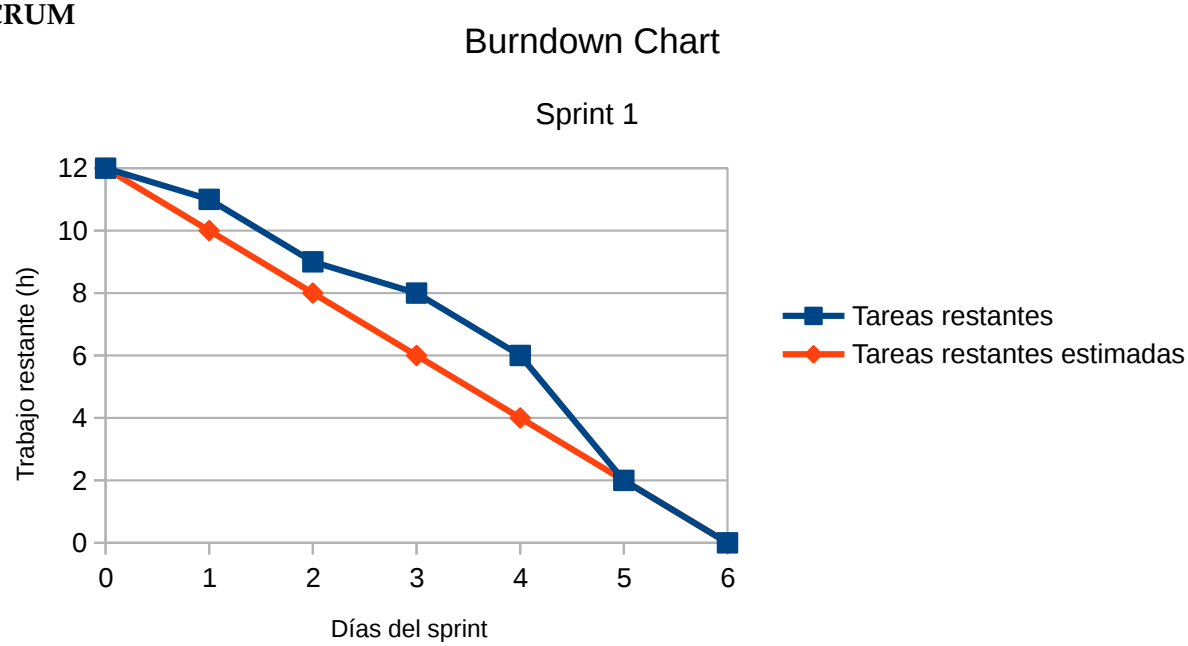
- ID.
- Nombre.
- Prioridad (importante: indicar escala).
- Puntos estimados (número de horas que consumirá implementar esa funcionalidad).
- Responsable de la implementación.
- Descripción.
- Validación.

SCRUM

Sprint backlog. Deberá aportar la siguiente información:

- Tareas pendientes.
- Tareas en curso.
- Tareas completadas.

SCRUM



SCRUM

Se realizan las siguientes reuniones:

- **Sprint Planning Meeting**: se realiza al inicio del sprint para organizar las tareas que se van a realizar durante el mismo.
- **Reuniones diarias** para informar del progreso, problemas encontrados y futuros pasos.
- **Sprint Review** y **Sprint Retrospective**: se realizan al final del sprint para controlar los progresos y compartir conocimientos.

0.2. Taiga.io

Taiga.io

- Herramienta online gratuita que permite organizar el desarrollo de un proyecto usando la metodología SCRUM, entre otras.
- Disponible en `https://taiga.io/`
- Permite crear el Product Backlog, los Sprint Backlogs y los Burndown Charts de manera sencilla.

0.3. Restricciones

Restricciones

- El proyecto se realizará haciendo uso de la herramienta Taiga.io.
- Se realizará un sprint por cada semana de trabajo.
- Para seguir la metodología SCRUM, el *scrum master* debe dividir el trabajo entre los miembros del equipo, otorgando una cantidad de tiempo a cada tarea.
- Cada miembro del equipo debe hacer la funcionalidad otorgada e implementarla. Puede haber un programador, un tester, etc.

1. Implementación y pruebas

Implementación

- Tal y como se comentó en la primera práctica, haremos uso del lenguaje C++.
- En caso de querer utilizar un IDE, dispondremos de Eclipse.
- El código debe ser legible y estar comentado correctamente, para una adecuada interpretación.

Pruebas

- Siguiendo la metodología SCRUM, debemos detectar posibles fallos y proponer soluciones.
- Este flujo de acciones debemos realizarlas de forma incremental (semanalmente).
- Los fallos deben estar bien documentados, así como las posibles soluciones.

2. Técnicas de validación

Matriz de trazabilidad

- Matriz requisitos funcionales (RF) - casos de uso (CU): cada RF debe quedar cubierto por al menos un CU. Con esto nos aseguramos que todas las funcionalidades requeridas son tenidas en cuenta.
- Matriz casos de uso (CU) - clases: cada caso de uso debe tener asignada una clase al menos, en caso contrario, faltaría mejorar la definición de la clase, o la creación de otra.

3. Entregables

3.1. SCRUM

Entregables SCRUM

1. Se deberá entregar en la tarea de Moodle el enlace al proyecto de Taiga.io junto con los nombres de los miembros del grupo.
2. La entrega se realizará antes de haber finalizado la práctica por completo con el objetivo de que los profesores podamos ver la dinámica de trabajo del grupo.
3. **Fecha límite de entrega en Moodle: 22 de noviembre a las 23.59h.**
4. La práctica **se evaluará el día 29 de noviembre**, fecha para la que se espera que se hayan completado dos sprints. Esta evaluación tendrá como objetivo dar retroalimentación a los alumnos.

3.2. Final

Entregable final

1. Software implementado correctamente documentado.
2. Proyecto de Taiga.io con el resto de sprints que se hayan realizado.
3. Documentación técnica, profesional y formal, con todas las prácticas incluidas en él (incluir las técnicas de validación). Véase el documento "Descripción general de las prácticas" en Moodle para más detalles.

Fecha límite de entrega: semana del 16 al 22 de diciembre.

4. Referencias

Referencias

Referencias

[Arlow] Arlow, J. y Neustadt, I. (2016). Programación UML 2. Programación (Anaya Multimedia) Anaya Multimedia.

[Debrauwer] Debrauwer, L., y Van der Heyde, F. (2016). UML 2.5: iniciación, ejemplos y ejercicios corregidos. Ediciones ENI.

[Miles] Miles, R., y Hamilton, K. (2006). Learning UML 2.0. O'Reilly Media, Inc. 2