

	Proceso de Software y Gestión 2 Análisis de diferencias entre el sprint S2 y S3 G2-24

**Universidad de Sevilla**  
**Escuela Técnica Superior de Ingeniería Informática**



**Grado en Ingeniería Informática – Ingeniería del Software**  
**Proceso de Software y Gestión 2**

Curso 2020 – 2021

## **ANÁLISIS DE DIFERENCIAS ENTRE EL SPRINT S2 Y S3**

Grupo de prácticas	G2-24
Autores	Rol
Rodríguez Pérez, Francisco	Scrum Master
Colmenero Capote, Pablo	Scrum Team Member
Martin Núñez, Ángel	Scrum Team Member
Barragán Salazar, David	Scrum Team Member
Pastor Fernández, Ginés	Scrum Team Member
Müller Cejas, Carlos Guillermo	Product Owner



## Diferencias de métricas y análisis (S2 – S3)

Como se puede observar en los gráficos **BurnDown** (véase en el documento A-3.3) de ambos sprints, hay diferencias en el trabajo acumulado durante el sprint. En sprint 2, el equipo, salvo en ciertos puntos no está tan alejado de la línea gris, que marca el devenir ideal de las tareas. No había mucho trabajo acumulado y las tareas se llevaban más o menos al día. Al final del sprint se puede ver en el gráfico que el equipo se acerca mucho a la línea gris. Además, los porcentajes de “Story Points” y “Pull Request” reflejan que casi todas las tareas fueron completadas.

Sin embargo, en el sprint 3 se puede observar que con el pasar de los días, el equipo arrastra mucho trabajo acumulado. A partir del 23 de abril, ese pico baja un poco, y no es hasta el 29 de abril cuando realmente empiezan a completarse la gran parte de las tareas, cuando el sprint estaba finalizando. A diferencia del sprint 2, el equipo nunca llega a acercarse demasiado a la línea gris, además se puede observar en los porcentajes de “Story Points” y “Pull Request” que quedan tareas muchas más tareas por realizar que en el sprint 2.

La principal diferencia es el trabajo que el equipo ha dejado acumulado. Dicha diferencia reside en que durante el sprint 2, el equipo estaba más al día con el trabajo a realizar, sin embargo, en el sprint 3, el trabajo se fue dejando con las semanas y no fue hasta el final del sprint cuando el equipo comenzó a realizar todas las tareas que requería dicho sprint.

Con respecto a los análisis realizados a partir de **SonarCloud** (véase en el documento A-3.4), una plataforma de análisis de código continuo que permite analizar el código del proyecto, se puede observar que en el sprint 2 hay un total de 18 bugs que resaltan problemas de seguridad y califica con D la vulnerabilidad, mostrando 13 puntos que pueden causar problemas de seguridad. Apenas hay deuda técnica (la cual califica con A), indica que existen 48 test unitarios y un porcentaje menor al 1% de código duplicado.

En el sprint 3, al realizar este mismo análisis, las distintas métricas muestran resultados similares al sprint 2. Los bugs se incrementan a 26 aunque la calificación es la misma, la vulnerabilidad sigue siendo calificada con E, los puntos de acceso al igual que en el sprint anterior son 2, tampoco hay una gran cantidad de deuda técnica, por lo que apenas se generan intereses, los tests unitarios caen a un 26% y las duplicaciones aumentan en un 1,9%.

Los resultados obtenidos de las métricas no han aumentado demasiado teniendo en cuenta la cantidad de código implementado. Al haber aumentado la cantidad de código, tanto bugs potenciales como el código duplicado ha aumentado su valor. Con respecto a la seguridad y vulnerabilidad, el resultado es casi el mismo, pues el equipo ha trabajado de la misma forma que en el sprint anterior. Sin embargo, el porcentaje de tests ha bajado casi la mitad con respecto al sprint 2 ya que el equipo no ha realizado ningún test en el presente sprint debido a la falta de tiempo y al no ser obligatorios, el equipo no se ha enfocado demasiado en ese aspecto.