

# Desarrollo de una versión mejorada del sistema AGATA de la ESPOL

## PROBLEMA

La Facultad de Ingeniería en Electricidad y Computación (FIEC) de ESPOL ha desarrollado un sistema llamado AGATA (Aplicación web para la gestión de Archivos de titulación Agata), que no ha sido sujeto a pruebas de software adecuadas, por lo que no se conoce con exactitud, si el flujo de cada proceso en los módulos tutor, profesor, secretaria y subdecanato está funcionando correctamente.

## OBJETIVO GENERAL

Demostrar el correcto funcionamiento de los módulos tutor, profesor, secretaria y subdecanato del sistema AGATA aplicando un plan de pruebas de software.



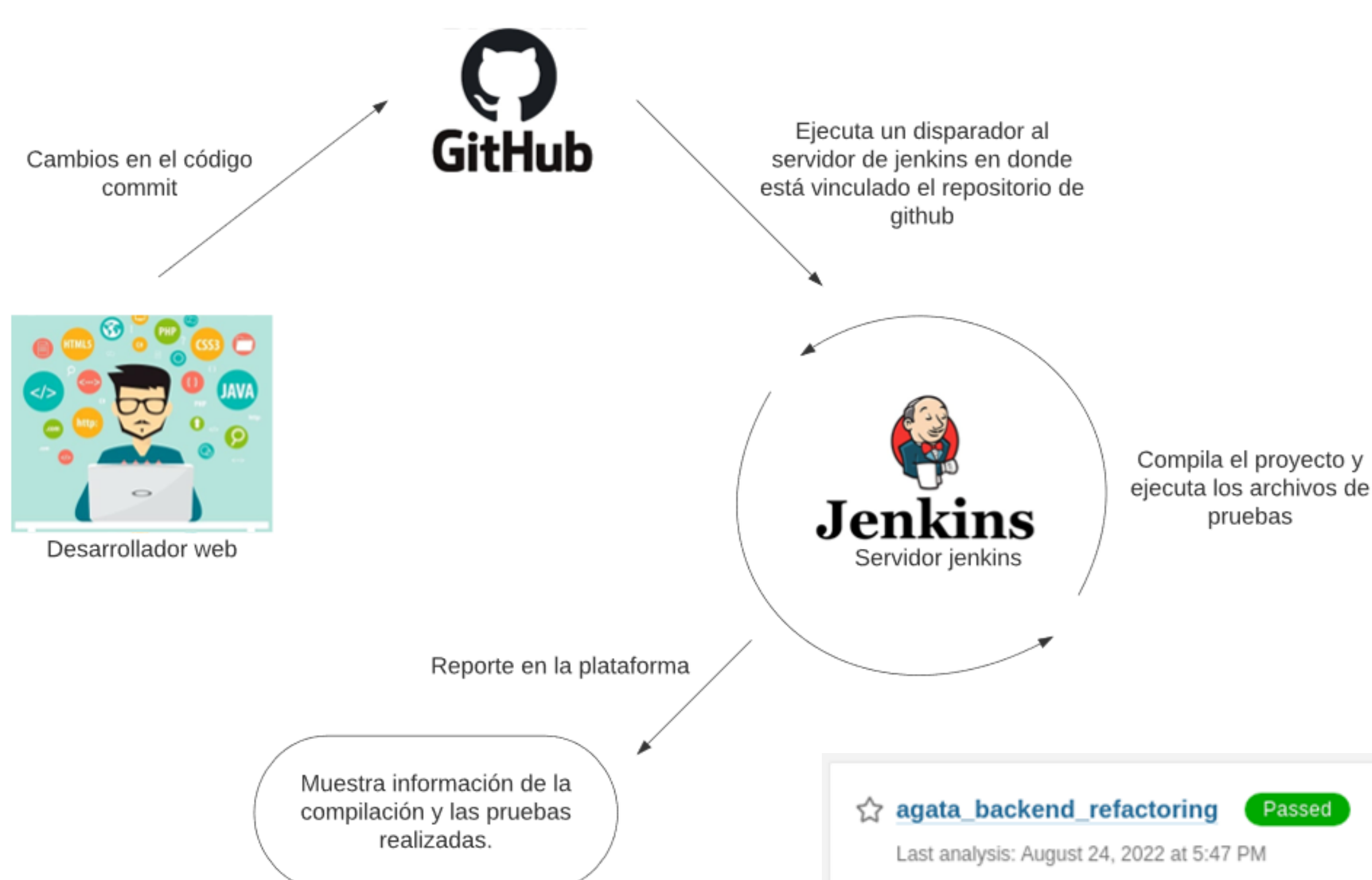
## PROPUESTA

Desarrollar y ejecutar pruebas unitarias e integración sobre los módulos tutor, profesor, secretaria y subdecanato del sistema AGATA, con la finalidad de detectar errores presentes y corregirlos. Así mismo evaluar el código desarrollado con la finalidad de mejorar la calidad del mismo.

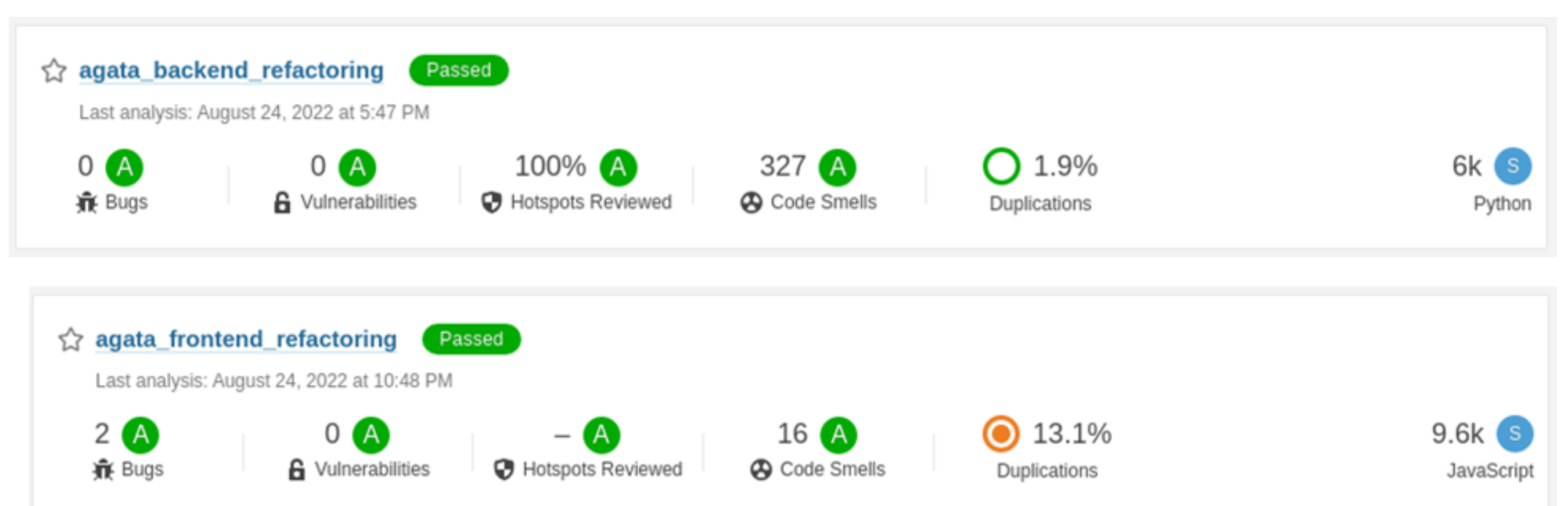


## RESULTADOS

Se realizaron en total 98 pruebas, 74 unitarias y 24 de integración, en la que se pudo conocer que las funciones empleadas tanto en el BackEnd como el FrontEnd están ejecutando correctamente sus procesos, también las pruebas desarrolladas quedaron automatizadas gracias al uso de la herramienta Jenkins y la calidad del código mejoró gracias al análisis del código que se realizó y a la aplicación de la técnica de refactoring.



Pruebas unitarias				
BackEnd	FrontEnd	Aprobadas	Fallidas	Total
34	40	74	0	74
Pruebas de integración				
BackEnd	FrontEnd	Aprobadas	Fallidas	Total
24	0	24	0	24



## CONCLUSIONES

- El desarrollo de pruebas de software tanto unitarias como de integración en el BackEnd y el FrontEnd permitió comprobar el correcto funcionamiento de los módulos tutor, profesor, secretaria y subdecanato dentro del sistema Agata.
- La herramienta Jenkins permitió que las pruebas de software se ejecuten de manera automática, lo que a futuro permitiría detectar errores rápidamente durante el desarrollo de nuevas funcionalidades justo después de un commit.
- El análisis del código utilizando la herramienta SonarQube aplicado al BackEnd y el FrontEnd permitió identificar porciones del código que necesitaban ser refactorizadas y una vez concluido el proceso de refactoring, la calidad del código mejoró en ambos repositorios.