

**INFORME TÉCNICO DE PLAN DE TRABAJO PARA CONSTRUCCIÓN
DE SOFTWARE**

EVIDENCIA – GA7-220501096-AA1-EV01

Presentado por:

Lady Fernanda Quintero Martínez.



Sistema optimizado para la formación integral y el
aprendizajeTecnología Análisis y desarrollo de software
modalidad virtual

Ficha 2879649

16 febrero 2025

1. Introducción

El desarrollo fuente del código es vital para el proceso de integración continua (CI). El versionamiento de código nos permitiera mantener la versión de cambios, colaborar eficientemente, y así minimizar riesgos en la implementación de nuevas funciones.

2. Objetivo

El objetivo de este informe es poder identificar y seleccionar las herramientas u las tecnologías de versionamientos que se integrarán en el flujo de trabajo de desarrollo de software “Cotización de gráficas Boyacá”, así se asegura un control eficiente de cambios, colaboración entre desarrolladores y estabilidad en el código fuente

3. Selección de herramientas de versionamiento

A continuación se presentan las herramientas de versionamiento que se usará, según la lectura de la guía 7 sobre integración continua.

3.1. Sistemas de control de versiones: Git

- Es el sistema de control de versiones más usado en esta industria
- Permite un historial de cambios al detalle y con posibilidad de revertirlos a versiones anteriores
- Facilita trabajar en equipo por medio de ramas y así no se afecta la master

3.2. Plataforma de repositorio: Git Hub

- El código fuente se alojara de manera confiable y escalable
- Se podrá gestionar, rastrear y registrar cambios, integrando herramientas de revisión, control y gestión de proyectos.
- Es compatible con grupos de trabajo de integración continua y entrega continua de (CI/CD)

3.3. Automatización de integración continua : Git Hub Actions

- Permite automatizar pruebas y despliegues con workflows que se pueden configurar, según tus necesidades
- Se integra direct con Git Hub, así facilita la compilación o construcción de pruebas automáticas
- Es compatible con una variedad amplia de lenguajes de programación y entornos de despliegue.

3.4. Administración de dependencias y versiones:

Dependabot

- Actualiza y monitorea automáticamente las dependencias del proyecto
- Mejora la seguridad del software al alertar sobre vulnerabilidades en las librerías usadas.

4. Conclusión

Al usar herramientas de Git Hub como almacenamiento, Git Hub actions para automatizar el CI/CD se puede garantizar un flujo de trabajo muy eficiente y seguro, para el desarrollo del software “Cotización de Gráficas Boyaca”

La integración de Dependabot contribuye a mantener actualizadas y seguras las dependencias del proyecto.

Estas herramientas permiten mejorar la calidad del código, reducir los riesgos y optimizar la colaboración en el equipo de desarrollo, asegurando la entrega de un producto robusto y confiable.

5. Referencias

- Sena, 2021. Integración Continua . *Zajuna.edu.co* , contenidos.
- Chacon, S., & Straub, B. (2014). *Pro git* (p. 456). Springer Nature.
- Pai, A. (2021). Gitraffe: Improving the GIT SCM Experience Through Revamped Interactions and Intent Classification.
- Nikolaev Hristozov, N. (2024). Automatización de la corrección de ejercicios de programación utilizando Github Actions.