informe técnico de plan de trabajo para construcción de software

Por:

Valentina Golu Martinez

Servicio Nacional de Aprendizaje -SENA- Tecnología en Análisis y Desarrollo de Software Ficha: 2882847

Instructor: Jhon Nuñez

cali, 23 de febrero de 2025

**INTRODUCCION**

En el siguiente documento hablaremos sobre la integracion continua es una practica de desarrollo de software que se ha vuelto cada vez mas popular en los ultimos años. Esta practica implica la automatizacion del proceso de construccion, prueba y despliegue de software, lo que permite a los equipos de desarrollo trabajar de manera mas eficiente y rapida. En esencia, la integracion continua implica la integracion de codigo fuente y la ejecucion de pruebas automatizadas para detectar errores temprano en el proceso de desarrollo.

**OBJETIVO**

El principal objetivo de esto es elegir una herramienta de versionamiento sobre la cual se va a trabajar a futuro, lo cual nos ayudara a futuro en el proceso a desarrollar y la facilidad que nos proporciona a la hora de crear los distintos repositorios que nos permitan desarrollar una solucion practica para futuras versiones y evitar alguna perdida de informacion importante.

**SELECCIÓN DE HERRAMIENTA DE VERSIONAMIENTO A UTILIZAR:**

De acuerdo con el material de estudio proporcionado, para el proyecto de elegiria Gitlab es una plataforma de desarrollo de software que proporciona un conjunto de herramientas para la gestion del ciclo de vida de desarrollo de aplicaciones, a continuacion, se presentan algunas de las caracteristicas claves de GITLAB:

**Propiedad:** Gitlab es propiedad de Gitlab Inc.

**Modelo de Licencia:** Gitlab es de codigo abierto y ofrece tanto uan version gratuita como una version de pago con mas funcionalidades.

**Funcionalidades Adicionales:** Gitlab ofrece una amplia gama de funcionalidades adicionales, como integracion continua, orquestacion de contenedores, gestion de permisos y mas.

**Hosting:** Gitlab puede ser instalado en los propios servidores a usar o en la nube.

**Control de versiones basado en Git:** Gitlab utiliza el sistema de control de versiones Git, lo que permite a los equipos de desarrollo rastrea y gestiona cambios en el codigo fuente de manera eficiente.

**Repositorios y ramas:** permite crear repositorios para alojar el codigo fuente y trabajar en diferentes ramas para desarrollar nuevas caracteristicas o solucionar problemas sin afectar la rama principal (generalmente “master” o “main”.

**Integracion continua (CI):** Gitlab ofrece integracion continua integrada, lo que significa que puedes automatizar la construccion, pruebas y despliegue de tu aplicación en diferentes entornos cada vez que se realiza un cambio en el repositorio.

**Registro de contenedores:** ofrece un registro de contenedores integrado para almacenar y distribuir imágenes de docker u otros contenedores. Esto facilita el proceso de implementacion y escalado de aplicaciones en contenedores.

**Seguridad y pruebas de seguridad:** ofrece heramientas de analisis estatico y dinamico de codigo para identificar problemas de seguridad y vulnerabilidad en el codigo. Tambien se integra con herramientas externas de analisis de seguridad.

**CONCLUSION**

El uso de Giltlab puede ser muy beneficioso para los equipos de desarrollo de software, ya que ofrece una gran cantidad de funcionalidades que pueden facilitar el trabajo colaborativo y la automatizacion de tareas.