

## **Informe Técnico De Plan De Trabajo Para Construcción De Software**

**Planear Actividades De Construcción Del Software De Acuerdo Con El Diseño Establecido**

**Aprendiz:**

**Santiago Carranza Carrillo**

**Ficha:**

**2848530-A**

**Instructor:**

**Julio Roberto Galvis Cardozo**

**SENA**

**Centro De Diseño Y Metrología**

**30 De septiembre 2024**

## **Contenido**

<b>1. Introducción .....</b>	<b>3</b>
<b>2. Objetivos.....</b>	<b>3</b>
<b>3. Selección de herramientas de versionamiento.....</b>	<b>3</b>
<b>3.1 Sistemas de control de versiones .....</b>	<b>3</b>
<b>4. Plataformas para implementar integración continua - Gitlab.....</b>	<b>5</b>

## **1. Introducción**

En el siguiente documento se presentará un informe técnico el cual expone según las características del software a desarrollar, las herramientas y tecnologías de versionamiento a utilizar en base a los conocimientos recolectados tras la visualización de los componentes presentados en el documento “Integración Continua”, Se presentarán y explicarán las razones por las cuales estas herramientas se consideradas óptimas y oportunas para el futuro desarrollo del software manteniendo esto en base al documento mencionado.

## **2. Objetivos**

- Definir las herramientas y tecnologías de versionamiento a utilizar dentro del software a desarrollar.
- Exponer las razones por las cuales se considera correcto utilizar la herramienta y tecnología seleccionada.
- Desarrollar de manera satisfactoria el informe técnico y generar la evidencia solicitada dentro de la actividad.

## **3. Selección de herramientas de versionamiento**

### **3.1 Sistemas de control de versiones**

Cuando hablamos de un sistema de control de versiones nos referimos a el sistema que se encarga del registro de todo cambio que se realiza dentro de un archivo o un conjunto de estos mismos a lo largo del tiempo, de esta manera estos archivos pueden ser recuperados si es que es requerido por alguno de los usuarios partícipes, dentro de los sistemas de control de versiones podemos encontrar los sistemas de control de versiones locales, centralizadas y distribuidas.

- **Sistemas de control de versiones locales**

Se caracterizan por crear copias manuales dentro de un directorio exclusivo, que pueda ser reconocido de manera sencilla, en este se llevará el registro de los cambios que se realizan dentro de alguno de los documentos específicos esto por lo regular a través de una enumeración o implementando fechas de realización de la versión, sin embargo esto suele ser riesgoso ya que a la hora de ingresar los registros pueden ocasionar errores, algo tan simple como olvidar la dirección del archivo debido a una mala organización de los mismos o que el usuario sobrescriba algún archivo o lo nombre de manera incorrecta por accidente, además de su desventaja principal, los sistemas de control de versiones locales tienen un carácter individual a la hora de ser implementado en un entorno de varios usuarios esto genera inconvenientes al tener que distribuir los archivos de forma manual, generando problemáticas graves a la hora de poseer un archivo actualizado y correcto.

- **Sistemas de control de versiones centralizados**

Este sistema surge para solucionar los problemas asociados al control de versiones dentro de un grupo de trabajo, en estos se posee un servidor en el cual se almacenan todos los archivos de los cuales varios clientes pueden realizar descargas de los mismos, de la misma manera este tipo de sistema de control de versiones es muy popular actualmente debido a su facilidad para trabajar en grupo no obstante al ser un sistema centralizado su principal desventaja es la dependencia a un solo equipo central de manera que si este equipo presenta algún error se bloquee la conexión de todos los usuarios que deseen acceder a los archivos.

- **Sistemas de control de versiones distribuidas (Seleccionado)**

En este tipo de sistema los clientes no solo descargan una copia de los archivos que se presentan dentro del repositorio, también generan una réplica completa del mismo, de esta forma si el servidor presenta algún fallo durante su funcionamiento los repositorios que poseen los clientes se

utilizan para la restauración del mismo, de esta manera los clientes forman parte de un respaldo o clones del repositorio que posee el servidor.

Se seleccionó el sistema de control de versiones de tipo distribuidas gracias a diversos factores tales como permitir la administración de varios repositorios remotos con los que el usuario específico puede trabajar al mismo tiempo junto a grupos de personas en otros proyectos y aún más útil dentro del mismo proyecto pero de diferentes formas, de manera que implementan varios flujos de trabajo los cuales no se podrían aplicar dentro de un sistema centralizado y obviamente en un sistema local, además de la facilidad de restauración del servidor gracias a las copias del repositorio que posee cada uno de los usuarios o clientes es un factor muy significativo y clave para la selección de este sistema de control como el óptimo a la hora de desarrollar el software.

#### **4. Plataformas para implementar integración continua - Gitlab**

A la hora de comparar las plataformas que permitan la integración del sistema de versionamiento se comparó las herramientas y demás apartados significativos que estas ofrecían, en base a estos factores y conocimiento previo de la plataforma se optó por usar la plataforma **GitHub** esta es un portal para alojar código de aplicaciones, en la cual ya se había experimentado en ocasiones anteriores dentro del desarrollo del proyecto de formación.