

Introducción de Git, GitHub y GitLab

Oscar Yesid Grande Tibana
Estudiante de Ingeniería Electrónica
Universidad Santo Tomas

Resumen—Este documento presenta una breve descripción y comparación de las plataformas Git, GitHub y GitLab, incluyendo sus funciones, ventajas y diferencias clave, así como referencias oficiales para su estudio.

I. INTRODUCCIÓN

El control de versiones es una herramienta esencial en el desarrollo moderno de software. Git, GitHub y GitLab son tres tecnologías ampliamente utilizadas para gestionar proyectos, facilitar la colaboración y mantener la integridad del código.

II. GIT

Git es un sistema de control de versiones que registra los cambios realizados en uno o varios archivos, permitiendo modificar o recuperar versiones anteriores y detectar problemas [1]. Git maneja los datos como un conjunto de copias instantáneas de un sistema de archivos en miniatura, almacenando únicamente los cambios cuando estos ocurren.

Las modificaciones se gestionan localmente, sin necesidad de conexión a internet, y posteriormente se sincronizan con un repositorio remoto. La integridad de los datos se garantiza mediante un valor *hash* calculado por el algoritmo *checksum*, el cual produce una cadena de 40 caracteres hexadecimales, por ejemplo:

```
24b9da6552252987aa493b52f8696cd6d3b00042
```

III. GITHUB

GitHub es una plataforma en la nube para almacenar, colaborar y compartir proyectos de software utilizando Git como sistema de control de versiones [2]. Los repositorios pueden ser públicos o privados, permitiendo un control seguro y profesional del desarrollo.

IV. GITLAB

GitLab es una plataforma de gestión de repositorios Git que integra todo el ciclo de vida del desarrollo: planificación, codificación, pruebas, despliegue y monitoreo [3]. Además de ofrecer repositorios remotos, permite automatizar compilaciones, pruebas y despliegues mediante *pipelines*, manteniendo el código siempre listo para producción.

GitLab incorpora herramientas de seguridad como análisis estático (SAST) y dinámico (DAST) para la detección de vulnerabilidades. También ofrece métricas de productividad, eficiencia e innovación a través de un almacenamiento de datos unificado.

V. COMPARACIÓN Y VENTAJAS DE GITLAB

GitLab se considera más rápido en ciclos de desarrollo, reduciendo el tiempo dedicado al mantenimiento de herramientas y aumentando la productividad. Incluye soporte con inteligencia artificial, gestión de políticas de seguridad y estrategias multinube sin dependencia de proveedores [4].

VI. COMO CREAR TU REPOSITORIO(README)

Para crear tu primer repositorio, es facil ve directamente a tu navegador busca Github, crea tu cuenta e inicia sesión.

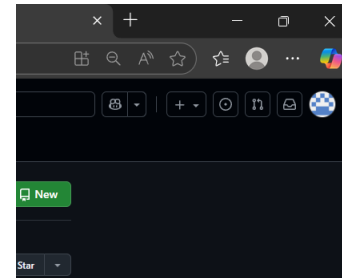


Figura 1. Visualizar perfil

Una vez inicies sesiona y demas, podrás ver en la pantalla de inicio de Github, en la esquina superior derecha una imagen tal vez un poco pixelada si no se ha insertado de lo contrario. Alli dara click y ingresara a...

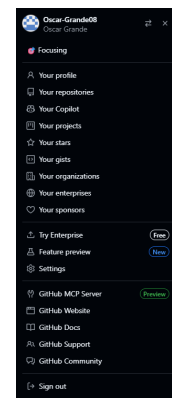


Figura 2. Menú de tu perfil

una vez alli se podra ingresar en bastantes opciones pero para crear repositorio, ingresa en "Tus repositorios".

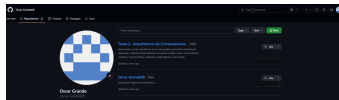


Figura 3. Menú repositorios

Una vez ubicado en el menu de tus repositorios, dale click en el recuadro verde "New o New Repository".

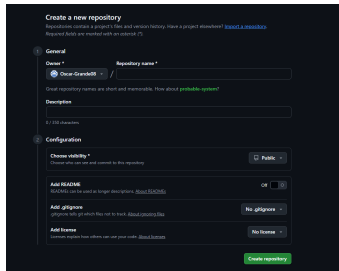


Figura 4. Crear Repositorio.

Una vez en este menú es de vital importancia que en el recuadro Repository name*, ingrese su mismo nombre de usuario. En visibilidad sea de carácter publica, y en el selector de activado o desactivado ADD README este sea activo, una vez finalizado estos pasos podrá dar en el recuadro verde de CREATE REPOSITORY y listo, encontrara su nuevo Readme en el cual podrá hacer una breve descripción de su vida personal como profesión, pasatiempos y algunos aprendizajes con los que cuente, los cuales puedan servir para su vida profesional y le puedan interesar a alguien mas. La idea es que de aquí parta y pueda realizar su trabajo personal de manera detallada y creativa, puede añadir imágenes o gifs, según sea el caso necesario o usted lo requiera, recuerde una vez mas es de su propiedad y puede personalizarlo como quiera.

REFERENCIAS

- [1] <https://git-scm.com/book/es/v2/Inicio--Sobre-el-Control-de-Versiones-Fundamentos-de-Git>
- [2] <https://docs.github.com/en/get-started/start-your-journey/about-github-and-git>
- [3] <https://about.gitlab.com/why-gitlab/>
- [4] <https://docs.gitlab.com>