



UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN

Unach
UNIVERSIDAD NACIONAL DE CHIMBORAZO
en movimiento

UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD DE INGENIERÍA



INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN

NOMBRE:

Elvis Santiago Pilco Paucar

FECHA: 2024/06/28

INTEROPERABILIDAD DE PLATAFORMAS

CONSULTA GITHUB

PERIODO ACADÉMICO 2024-1S

Capturas

1. ¿Qué es Github?



GitHub es un portal creado para alojar el código de las aplicaciones de cualquier desarrollador, y que fue comprada por Microsoft en junio del 2018. La plataforma está creada para que los desarrolladores suban el código de sus aplicaciones y herramientas, y que como usuario no sólo puedas descargar la aplicación, sino también entrar a su perfil para leer sobre ella o colaborar con su desarrollo.

Como su nombre indica, la web utiliza el sistema de control de versiones Git diseñado por Linus Torvalds. Un sistema de gestión de versiones es ese con el que los desarrolladores pueden administrar su proyecto, ordenando el código de cada una de las nuevas versiones que sacan de sus aplicaciones para evitar confusiones. Así, al tener copias de cada una de las versiones de su aplicación, no se perderán los estados anteriores cuando se va a actualizar.

Así pues, Git es uno de estos sistemas de control, que permite comparar el código de un archivo para ver las diferencias entre las versiones, restaurar versiones antiguas si algo sale mal, y fusionar los cambios de distintas versiones. También permite trabajar con distintas ramas de un proyecto, como la de desarrollo para meter nuevas funciones al programa o la de producción para depurar los bugs.

Las principales características de la plataforma es que ofrece las mejores características de este tipo de servicios sin perder la simplicidad, y es una de las más utilizadas del mundo por los desarrolladores. Es multiplataforma, y tiene multitud de interfaces de usuario.

2. Características Principales

GitHub es una plataforma poderosa para el desarrollo colaborativo de software. Aquí algunas de sus características principales:

Control de versiones: Basado en Git, permite un seguimiento detallado de los cambios en el código.

Repositorios: Espacios donde se almacenan proyectos, con funciones como branches y forks.

Pull Requests: Permiten a los desarrolladores revisar y discutir cambios antes de fusionarlos.

Issues: Sistema de seguimiento de problemas y tareas.

GitHub Actions: Automatización de flujos de trabajo, como pruebas y despliegue continuo.

Wikis: Documentación colaborativa para cada repositorio.

Pages: Publicación de sitios web estáticos directamente desde un repositorio.

Proyectos: Tableros tipo Kanban para gestionar tareas y proyectos.

Colaboradores: Gestión de permisos y roles en los proyectos.

Integraciones: Soporte para numerosas aplicaciones y servicios de terceros.

Releases: Administración de versiones de software con notas de lanzamiento.

Code Review: Comentarios y discusiones en el código.

Markdown Support: Formato de documentos usando Markdown para README y otros archivos.

GitHub Copilot: Asistente de programación impulsado por IA.

Dependabot: Herramienta para gestionar dependencias y alertas de seguridad.

Graphs: Visualizaciones de estadísticas y actividad del proyecto.

Community Insights: Análisis de la contribución y participación en proyectos de código abierto.

GitHub Mobile: Aplicación para gestionar repositorios y notificaciones desde dispositivos móviles.

3. ¿Qué son los árboles en Github?

En GitHub, los árboles se refieren a la estructura de directorios y archivos dentro de un repositorio.

Cada árbol representa un directorio, y cada nodo en el árbol puede ser un archivo o un subdirectorio.

Esto permite organizar el contenido del repositorio de manera jerárquica.

Características de los árboles en GitHub:

Estructura Jerárquica: Organiza archivos y carpetas.

Commits: Cada commit guarda un estado del árbol.

Branches: Diferentes ramas pueden tener distintos árboles.

Historial: Permite ver cómo cambia el árbol a través del tiempo.

Visualización: GitHub permite navegar por el árbol del repositorio.

Los árboles son fundamentales para navegar y entender la organización del código en un proyecto.

4. Pasos para crear un repositorio

4.1 Iniciar sesión en GitHub:

Ve a github.com y accede a tu cuenta.

Ir a la página de repositorios:

En la esquina superior derecha, haz clic en el ícono de tu perfil y selecciona "Your repositories".

4.2 Crear un nuevo repositorio:

Haz clic en el botón "New" (Nuevo) o "Create repository" (Crear repositorio).

Configurar el repositorio:

Nombre del repositorio: Ingresa un nombre único.

Descripción (opcional): Añade una descripción breve del proyecto.

Visibilidad: Elige entre público o privado.

Opciones adicionales:

Initialize this repository with a README: Marca esta opción si deseas añadir un archivo README inicial.

.gitignore: Puedes seleccionar una plantilla para ignorar archivos específicos.

License: Selecciona una licencia para tu proyecto.

Crear el repositorio:

Haz clic en el botón "Create repository".

Después de crear el repositorio:

Clonar el repositorio:

Copia la URL y usa git clone en tu terminal:

bash

Copiar código

git clone <URL-del-repositorio>

Agregar archivos y hacer commits:

Añade tus archivos, realiza cambios, y usa git add y git commit para guardar los cambios.

Empujar cambios al repositorio:

Usa git push para subir tus cambios al repositorio en GitHub.

Bibliografía

Yúbal Fernández. (2019, October 30). *Qué es Github y qué es lo que le ofrece a los*

desarrolladores. Xataka.com; Xataka Basics. <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>

Líderes en formación tecnológica, reskilling y upskilling / OpenWebinars. (2019).

OpenWebinars.net. <https://openwebinars.net/blog/que-es-github/>