1 Taller GIT

- Objetivo: Proporcionar a los estudiantes conocimientos teóricos y prácticos sobre GIT, habilitándolos para utilizar eficazmente esta herramienta en entornos de desarrollo colaborativo. Al final del taller, los asistentes serán capaces de entender a plenitud los conceptos teoricos, iniciar proyectos y gestionar cambios.

- Secciones:

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1. Seccion Teorica:

Objetivo: Familizarse con conceptos clave de GIT, diferencia entre sistemas de control de versiones centralizados y distribuidos, commits, repositorios, importancia de GIT en la gestión de proyectos, introducción a branching y git Workflow.

Preguntas de seleccion multiple con unica respuesta:

1. ¿Qué es GIT?

A) Un lenguaje de programación.

B) Un sistema de control de versiones distribuido.

C) Un editor de texto.

D) Una red social para desarrolladores.

Tu respuesta:B

2. ¿Cuál es una ventaja de utilizar un sistema de control de versiones distribuido como GIT en comparación con uno centralizado?

A) Requiere más espacio en el servidor.

B) Es más complejo de usar.

C) Permite a los usuarios trabajar de forma independiente y fusionar cambios.

D) Solo el administrador puede hacer cambios.

Tu respuesta:C

3. Los 'commits' en GIT se pueden describir mejor como:

A) Versiones del software que están en producción.

B) Fotografías del estado de un proyecto en un momento dado.

C) Ramas nuevas en el repositorio.

D) Mensajes entre colaboradores.

Tu respuesta:B

4. Seleccione una razón por la que GIT es ampliamente utilizado:

A) Su capacidad para almacenar imágenes de alta resolución.

B) La habilidad de rastrear cambios y colaborar en proyectos.

C) Su integración con redes sociales.

D) La posibilidad de ejecutar código directamente desde el repositorio.

Tu respuesta:B

5. ¿Qué es un repositorio en GIT?

A) Una colección de documentos.

B) Un espacio de almacenamiento en la nube.

C) Un directorio de trabajo donde se almacenan los archivos del proyecto.

D) Un lugar donde se almacenan y gestionan las versiones de un proyecto.

Tu respuesta:D

6. ¿Cuál es la importancia de git en relación con la vieja forma de guardar archivos como 'final\_final\_este\_si.docx'?

A) Permite una mejor organización de las imágenes del proyecto.

B) Facilita la colaboración en documentos de texto.

C) Mejora la gestión de versiones, evitando la confusión de nombres de archivos.

D) Aumenta la velocidad de acceso a los archivos.

Tu respuesta:C

7. Para verificar la versión de GIT instalada en tu sistema, usarías el comando:

A) git --version

B) git version

C) git info

D) git check

Tu respuesta:A

8. Si deseas configurar tu nombre de usuario y email en GIT de forma local, ¿cuál de los siguientes comandos utilizarías?

A) git config --global user.name "Tu Nombre". git config --global user.email "Tu email".

B) git config --system user.name "Tu Nombre". git config --system user.email "Tu email".

C) git setup --local user.name "Tu Nombre". git config --system user.email "Tu email".

D) git initialize user "Tu Nombre". git config --global user.email "Tu email".

Tu respuesta:B

9. ¿Cuál de los siguientes comandos se utiliza para iniciar un nuevo repositorio GIT?

A) git start

B) git create

C) git init

D) git new

Tu respuesta:C

10. Para agregar cambios al área de preparación (staging area) en GIT, usarías:

A) git add .

B) git commit

C) git save

D) git stage

Tu respuesta:A

11. ¿Cuál es el propósito principal del workflow en GIT, desde el directorio de trabajo hasta el repositorio local?

A) Permitir el almacenamiento de backups en la nube.

B) Facilitar la edición de código directamente desde el repositorio.

C) Manejar el flujo de cambios a través de diferentes etapas hasta que se consolidan en el repositorio.

D) Configurar automáticamente los permisos de usuario y seguridad.

Tu respuesta:C

12. ¿Qué problema resuelve la normalización de fin de línea (CRLF a LF) en GIT, especialmente en Windows?

A) Previene conflictos en la codificación de caracteres.

B) Mejora la velocidad de las operaciones de GIT.

C) Evita problemas en la comparación y fusión de archivos debido a diferencias en los caracteres de fin de línea entre sistemas operativos.

D) Aumenta la seguridad al transferir archivos entre diferentes sistemas operativos.

Tu respuesta:C

13. En el contexto de GIT, ¿qué representa una rama?

A) Una copia exacta del repositorio central para cada colaborador.

B) Un cambio directo en el archivo en el repositorio principal.

C) Una versión del repositorio que diverge del flujo principal de trabajo, permitiendo el desarrollo paralelo sin afectar el estado principal.

D) Una herramienta para fusionar automáticamente cambios de diferentes colaboradores.

Tu respuesta:C

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. Seccion Practica: 🚀 Misión GIT-Galáctica

🌟 Objetivo de la mision: Como nuevo recluta de la Academia de Código Intergaláctico, tu primera misión es dominar los fundamentos de GIT, la herramienta esencial para todos los desarrolladores del espacio. Deberás configurar tu estación de trabajo, crear tu primer proyecto y aprender a gestionar tus cambios como un verdadero comandante de código.

🛠 Equipamiento Necesario: Acceso a una terminal bash o similar y GIT instalado en tu máquina espacial.

Instrucciones de la Misión:

0. 🌌 Checkeo del Espacio Exterior: Asegúrate de que GIT esté instalado en tu nave.

1. 📡 Configuración de la Base Espacial

1.1 🛠 Ajustes Iniciales: Abre la terminal y transmite tu identidad a GIT: agrega tu nombre y correo electronico de forma global y local.

1.2 ✅ Verifica que la configuracion haya sido aplicada correctamente.

2. 🚀 Creación de un Nuevo Repositorio

2.1 🌍 Nueva Base: Crea una carpeta llamada galactic\_project y navega hacia ella.

2.2 📄 Documentos Secretos: Genera dos archivos esenciales, index.html y styles.css.

2.3 🌟 Iniciando el Motor de GIT: Enciende los sistemas de control de versiones con git init.

3. 📦 Área de preparacion: Preparación para el Despegue: Coloca tus documentos secretos en el área de preparacion de GIT.

4. 🛸 Primer Vuelo: Documentando la Misión

4.1 📝 Bitácora de la Misión: Crea README.md con un mensaje para la posteridad:

# Learning Git with RIWI is fun!

4.2 🚀 Lanzamiento: Utilizando el comando git commit envía tu primera bitácora al repositorio de la misión con el siguiente mensaje: "Primer vuelo: Añadiendo README.md"

5. 🌠 Exploración del Cosmos: Avanzando hacia lo Desconocido

5.1 📄 Mensaje al Universo: Añade el esqueleto básico de HTML a index.html.

<!DOCTYPE html>

<html>

<head>

</head>

<body>

</body>

</html>

5.2 🔄 Estado de la Misión: Usa git status para inspeccionar el área de carga.

5.3 🌌 Nueva Frontera: Documenta tu exploración con un nuevo commit.

📚 Debriefing de la Misión:

Al regresar de tu viaje por el espacio GIT, presenta las capturas de tu terminal como pruebas de las hazañas alcanzadas. Cada captura es un testimonio de tu habilidad para navegar por el vasto universo de la gestión de versiones. ¡Buena suerte, comandante!

