

1. GIT – INTRODUCTION

GIT adalah sistem kontrol versi yang terdistribusi yang digunakan secara luas dalam pengembangan perangkat lunak. Ini adalah alat yang sangat berguna bagi pengembang untuk mengelola kode sumber proyek secara efisien, terutama dalam kolaborasi tim.

Berikut adalah beberapa poin penting dalam deskripsi tentang GIT:

1. Kontrol Versi Terdistribusi: GIT adalah sistem kontrol versi terdistribusi, yang berarti setiap pengguna memiliki salinan penuh dari repositori proyeknya sendiri. Setiap pengguna memiliki salinan penuh dari repositori proyeknya sendiri.
2. Repositori: Repositori GIT adalah tempat penyimpanan proyek perangkat lunak yang lengkap, termasuk semua versi file, riwayat perubahan, dan metadata lainnya.
3. Komit: Proses untuk menyimpan perubahan ke repositori disebut "komit". Setiap komit merekam snapshot dari semua file dalam proyek pada saat itu, bersama dengan metadata yang mencatat waktu, pengarang, dan pesan komit.
4. Cabang: GIT memungkinkan pengguna untuk membuat cabang dari repositori utama. Cabang ini memungkinkan pengembang untuk bekerja secara terisolasi pada fitur atau perbaikan bug tanpa mempengaruhi kode di cabang utama. Kemudian, cabang dapat digabungkan kembali ke cabang utama setelah selesai.
5. Konflik: Ketika dua atau lebih perubahan bersaing dalam repositori, konflik mungkin terjadi. GIT memungkinkan pengguna untuk menyelesaikan konflik ini secara manual dengan memutuskan perubahan mana yang akan diterima.

6. Riwayat Commit: GIT menyediakan riwayat lengkap dari setiap perubahan yang terjadi dalam repositori. Ini memungkinkan pengguna untuk melihat semua komit yang telah dibuat, bersama dengan perubahan yang dilakukan di setiap komit.
7. Kolaborasi Tim: GIT memfasilitasi kolaborasi tim dengan menyediakan alat untuk mengelola kode sumber bersama-sama. Ini termasuk penggunaan repositori bersama, penanganan cabang, dan mekanisme untuk memeriksa dan meninjau kode.

Dengan GIT, pengembang dapat bekerja secara efisien, melacak perubahan kode, berkolaborasi dengan tim, dan mengelola proyek perangkat lunak dengan lebih baik.