|  |
| --- |
| **CAK3EAB3 KOMPUTASI AWAN DAN TERDISTRIBUSI**  **Tugas-13**  **SaaS**  **Kelompok: xxx**  **KELAS IF-XX-XX**  **Dosen: XXX** |
| http://adam.staff.telkomuniversity.ac.id/wp-content/uploads/sites/10/2013/12/3.-Logo-Telkom-University-Konfigurasi-Memusat.png  **PROGRAM STUDI XXX**  **FAKULTAS INFORMATIKA**  **UNIVERSITAS TELKOM**  **BANDUNG**  **2024** |

**Ketentuan:**

1. Dikerjakan secara kelompok.
2. Dikerjakan langsung pada lembar soal.
3. Untuk soal perhitungan, maka cara atau langkah pengerjaan **wajib** dituliskan.
4. Setiap anggota kelompok wajib mengerjakan soal (ikut berkontribusi).
5. Anggota kelompok yang tidak ikut mengerjakan maka nilainya nol, nama mahasiswa tersebut tidak perlu ditulis di tabel.
6. Unggah jawaban ke LMS diwakili oleh salah satu anggota kelompok.
7. Lengkapi tabel pernyataan di bawah ini sebagai syarat nilai diinput.

**PERNYATAAN**

**Saya menyatakan bahwa saya:**

**(a) benar-benar ikut mengerjakan soal**

**(b) telah menjelaskan jawaban saya ke teman anggota kelompok**

**(c) telah memahami penjelasan dari teman anggota kelompok**

**(d) tidak mencontek jawaban dari kelompok lain**

**(e) tidak memberikan jawaban PR ini ke kelompok lain**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama** | **NIM** | **Kontribusi Tugas** | **Tanda Tangan** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**TUGAS 13**

1. **Carilah penyedia SaaS yang free!**
2. **Identifikasi masalah yang dihadapi (studi kasus)**
3. **Desainlah solusi menyelesaikan permasalahan tersebut**
4. **Gunakan salah satu layanan SaaS menyelesaikan permasalahan tersebut!**
5. **Penyedia SaaS Free untuk Version Controlling**

Penyedia SaaS yang menyediakan layanan version control gratis adalah GitHub. GitHub adalah platform yang menyediakan repository Git berbasis cloud, memungkinkan kolaborasi dan version control untuk proyek perangkat lunak. GitHub menawarkan paket gratis yang memungkinkan kita untuk membuat repository publik atau private dengan kapasitas penyimpanan yang cukup untuk kebanyakan proyek.

1. **Identifikasi Masalah yang Dihadapi (Studi Kasus)**

Masalah yang sering dihadapi dalam proyek pengembangan perangkat lunak adalah kesulitan dalam melacak perubahan kode (source code), berkolaborasi secara efektif antar anggota tim, serta mengelola versi-versi kode yang terus berubah. Hal ini bisa menyebabkan kesalahan, kehilangan kode, atau ketidaksesuaian antara versi yang digunakan oleh developer.

**Studi Kasus**: Sebuah tim pengembang perangkat lunak bekerja pada proyek bersama, dan mereka mengalami kesulitan dalam mengelola perubahan kode karena tidak ada sistem version control. Mereka kesulitan melacak revisi kode yang dilakukan oleh masing-masing anggota tim, dan beberapa kode yang telah diperbarui atau ditambahkan menyebabkan konflik ketika digabungkan.

1. Desain Solusi Menyelesaikan Permasalahan

Untuk menyelesaikan masalah ini, solusinya adalah menerapkan sistem version control berbasis cloud yang dapat:

* Menyediakan version control untuk melacak semua perubahan yang dilakukan terhadap kode sumber.
* Memungkinkan kolaborasi antar anggota tim, sehingga mereka dapat bekerja pada kode yang sama tanpa menyebabkan konflik.
* Mempermudah manajemen cabang (branching) untuk pengembangan fitur yang terpisah dan integrasi kode yang lebih baik.
* Memberikan kemudahan dalam mengelola dan menggabungkan kode melalui pull request dan merge.

Solusi ini akan menghilangkan masalah konflik kode, serta memungkinkan kolaborasi yang lebih efisien antara tim pengembang.

1. Penggunaan Layanan SaaS (GitHub)

Langkah-langkah untuk menyelesaikan permasalahan ini menggunakan GitHub:

* Buat akun GitHub: Tim perlu membuat akun di GitHub (gratis untuk penggunaan pribadi atau proyek open-source).

A screenshot of a computer

Description automatically generated

Gambar 1 Inisialisasi Repository di Github

* Buat Repository: Tim dapat membuat repository baru di GitHub untuk menyimpan hasil proyek yang kita kerjakan. Ini akan menjadi tempat penyimpanan utama kode.
* Inisialisasi Git di Local Machine: Setiap anggota tim menginisialisasi Git pada komputer lokal mereka dan menghubungkannya dengan repository GitHub.

A screenshot of a computer screen

Description automatically generated

Gambar 2 Inisialisasi Repository di Local

git init

git remote add origin <url\_repository\_github>

* Branching dan Commit: Setiap pengembang membuat cabang (branch) untuk fitur atau task mereka, melakukan commit secara berkala untuk menyimpan perubahan.

A screen shot of a computer program

Description automatically generated

Gambar 3 Membuat branch di Git

git checkout -b feature-xyz

git add .

git commit -m "Deskripsi perubahan"

* Pull Request (PR): Setelah selesai dengan perubahan di branch yang kita buat, anggota tim mengirimkan pull request di GitHub untuk menggabungkan perubahan ke branch utama (feature-xyz). Ini memungkinkan review kode dan meminimalkan konflik.

A screenshot of a computer

Description automatically generated

Gambar 4 List branch github

* Merge: Setelah pull request disetujui, perubahan digabungkan dengan branch utama.
* Penggunaan Issues: GitHub menyediakan fitur Issues untuk melacak bug atau task yang perlu diselesaikan. Ini membantu tim dalam mengorganisir pekerjaan mereka.

Dengan menggunakan GitHub, tim dapat lebih mudah mengelola kode, berkolaborasi secara efektif, dan meminimalkan konflik serta kehilangan data selama proses pengembangan.