



harisenin

# Bootcamp Full Stack Developer

Judul Mission : Mission 2  
Disusun oleh : Gerry Gerardo P - 12024  
Email : p.gerry03@gmail.com

Restricted - For Student Only

#JadiYangKamuMau





Dalam menyusun detail analisismu terhadap study case, pastikan komponen berikut ada dan sesuaikan pula urutan slidanya.

1. Identifikasi Masalah yang terdapat dalam study case (**Slide 1**)
2. Jelaskan bagaimana solusi penyelesaian dari masalah tersebut dengan 2 cara, yaitu :
  - **Slide 2** = Jabarkan menggunakan Diagram Alir/Flowchart
  - **Slide 3** = Jabarkan secara deskriptif

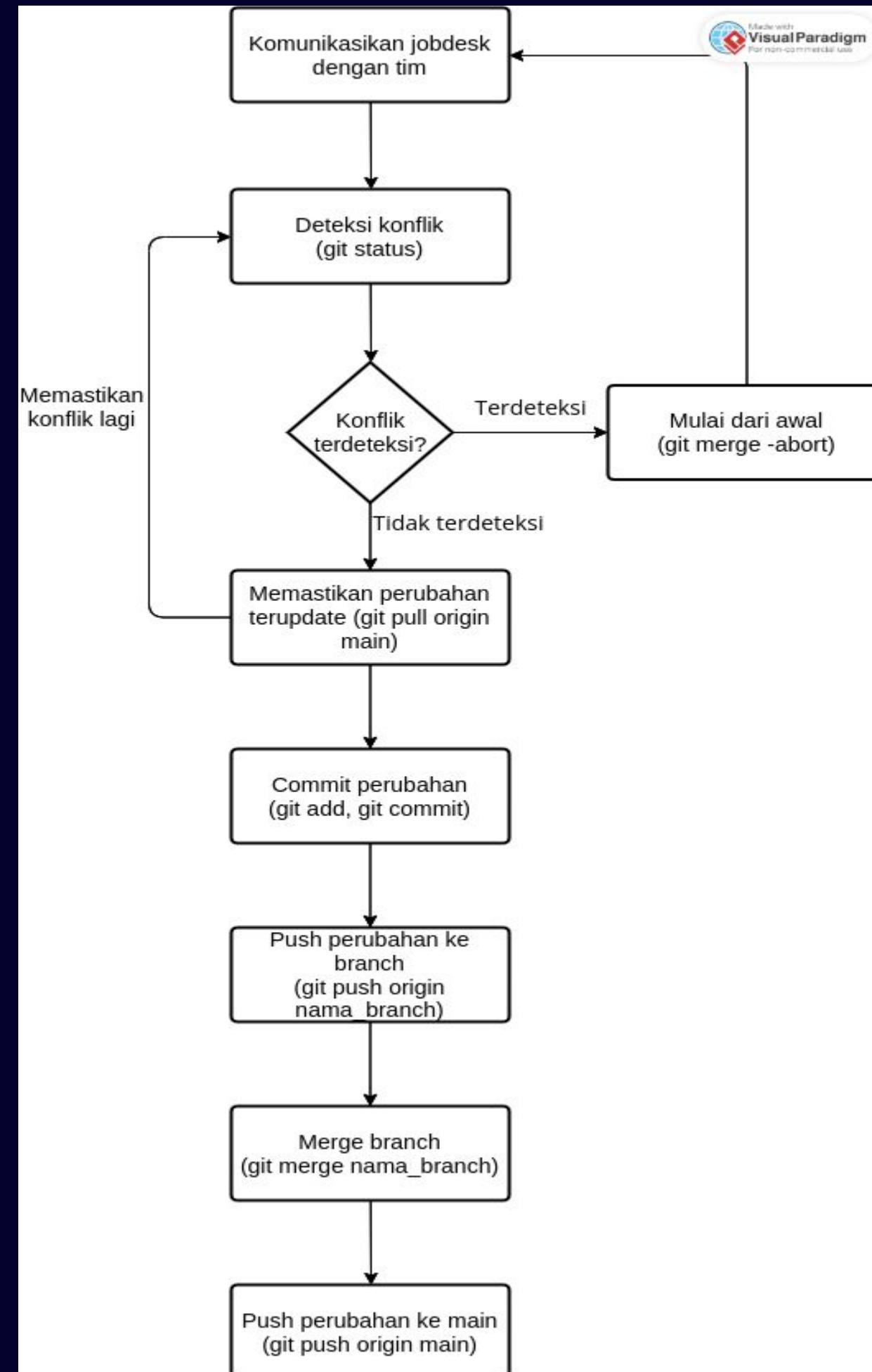


# Identifikasi Masalah Konflik Git

1. Pada masalah konflik git ini, terjadi karena kedua anggota tim melakukan perubahan pada 1 file yang sama, dimana dan juga di baris yang sama. Tentunya hal ini akan membuat git kebingungan dalam menentukan perubahan yang diinginkan.
2. Kedua tim tidak memiliki komunikasi dan pembagian tugas dan tanggung jawab yang jelas, hal ini terlihat dari masing-masing yang telah bekerja pada fitur yang berbeda, namun melakukan perubahan di satu file yang sama.
3. Ada beberapa step yang belum / tidak dilakukan oleh kedua orang tim tersebut saat push perubahan ke main branch.



# Flowchart Penyelesaian Git Conflict







# Penjelasan Flowchart

1. **Komunikasikan dengan tim**, pembagian jobdesk setiap orang dalam tim seharusnya dikomunikasikan dengan jelas, file, fitur, bagian apa saja yang harus dikerjakan masing-masing anggota tim, agar dapat meminimalisir konflik.
2. **Deteksi konflik**, saat perubahan pada file masing-masing branch telah dilakukan, dan ingin di push ke main branch, jalankan perintah ``git status`` terlebih dahulu, untuk melihat apakah ada perubahan yang menimbulkan konflik atau tidak. Jika terdeteksi konflik lakukan perubahan dari awal dan jalankan `git merge -abort`, lakukan kembali komunikasi dengan tim, jika tidak konflik maka perubahan terupdate dengan menjalankan ``git pull origin main``.



## Penjelasan Flowchart

3. **Lakukan kembali deteksi konflik**, setelah memastikan perubahan terupdate, lakukan kembali deteksi konflik dengan menjalankan perintah ``git status`` untuk memastikan apakah konflik terdeteksi kembali atau tidak. Jika iya maka lakukan step dari pertama kembali.
4. **Commit perubahan**, jika sudah dipastikan konflik tidak ada, dan perubahan sudah dipastikan terupdate, maka commit perubahan dengan menjalankan perintah ``git add``, kemudian ``git commit``.
5. **Push perubahan**, kini perubahan dari masing-masing branch dapat dipush ke masing-masing branch ``git push origin nama_branch``, kemudian ``git merge nama_branch`` dan ``git push origin main``.