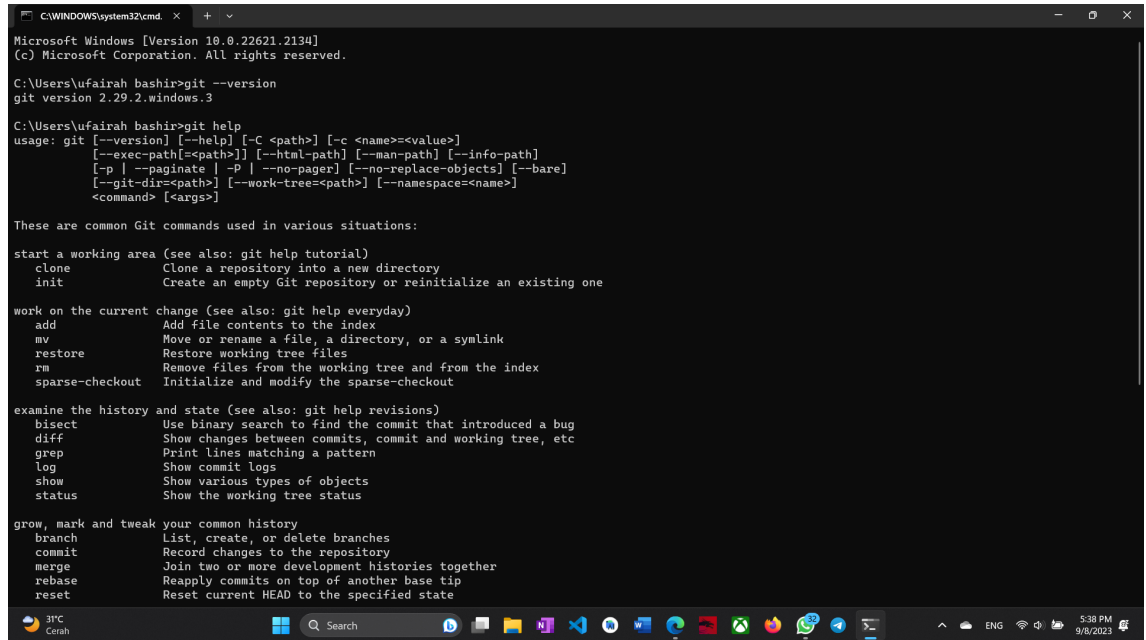


Nama: Ufairah Damara Bashir
Kelompok: 3
Universitas Hasanuddin | Sistem Informasi

Soal 1

1. Screenshoot Git



```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ufairah bashir>git --version
git version 2.29.2.windows.3

C:\Users\ufairah bashir>git help
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
      <command> [<args>]

These are common Git commands used in various situations:

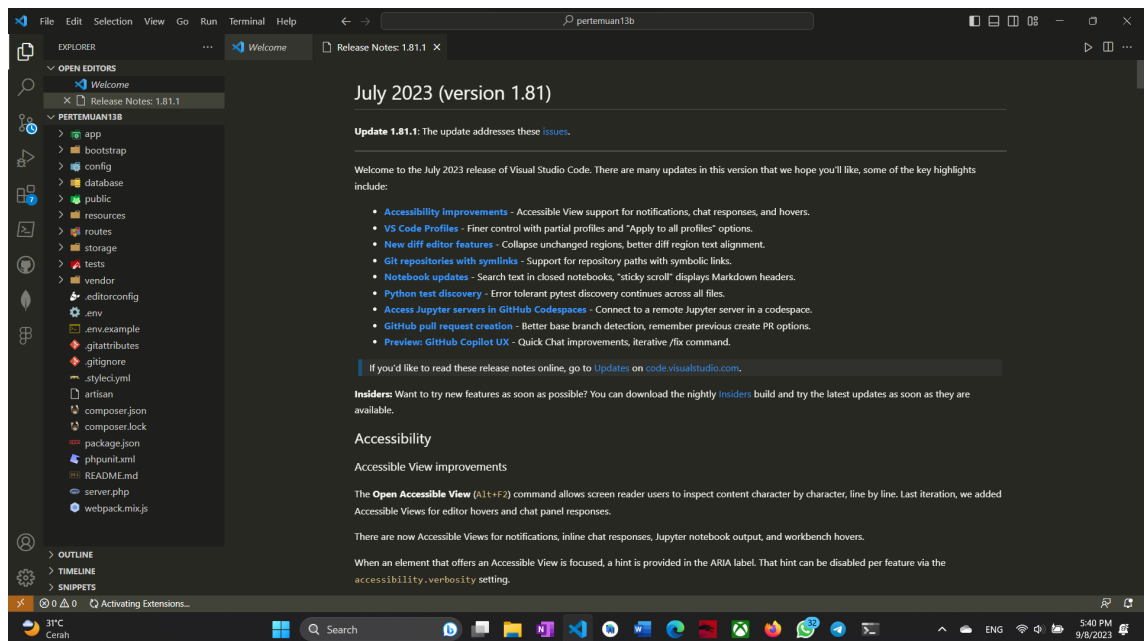
start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm         Remove files from the working tree and from the index
  sparse-checkout  Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  diff       Show changes between commits, commit and working tree, etc
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  branch     List, create, or delete branches
  commit     Record changes to the repository
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  reset      Reset current HEAD to the specified state
```

2. Screenshoot visual studio code



3. Summary:

a. SDLC (Siklus Hidup Pengembangan Perangkat Lunak)

SDLC adalah metodologi terdiri dari tahapan-tahapan yang digunakan untuk mengembangkan perangkat lunak. Tahapan-tahapan itu terdiri:

- Perencanaan dan analisis
Identifikasi masalah, menentukan ruang lingkup, alokasi sumber daya, waktu, pembagian tanggung jawab
- Desain
Merancang perangkat lunak dengan rinci meliputi arsitektur sistem, antarmuka pengguna (ui/ux) dan desain database
- Pengembangan
Mengimplementasikan rancangan yang telah didesain sebelumnya
- Pengujian
Pengujian produk perangkat lunak untuk memastikan fungsional kinerja, keamanan dan kualitas keseluruhan perangkat lunak
- Penerapan
Pengimplementasian di masyarakat
- pemeliharaan
Tetap menjalankan pemeliharaan untuk menjaga kualitas seperti meningkatkan fitur dan menjaga perangkat lunak

Manfaat SDLC

- Prediktabilitas dan pengendalian proyek
- Peningkatan kualitas perangkat lunak
- Pengelolaan risiko yang lebih baik
- Efisiensi tim dan kolaborasi
- Memenuhi kebutuhan pengguna
- Penghematan biaya dan waktu: menghindari perubahan mendadak
- Meningkatkan pengawasan dan evaluasi: evaluasi setiap tahap untuk kualitas yang lebih baik
- Peningkatan dokumentasi

Model model SDLC

- Waterfall : model ini linier berurutan artinya tidak akan ke tahap selanjutnya jika tahap yang sedang dikerjakan belum selesai. Tahapan ini mulai dari analisis, perencanaan, desain, pengembangan, pengujian, implementasi, dan pemeliharaan. Model ini cocok untuk proyek dengan persyaratan yang jelas dan stabil
- V-shape model: mirip seperti model waterfall tetapi setiap selesai satu tahapan akan dilakukan testing. Jika testing sudah memuaskan maka lanjut ke tahap berikutnya. Namun jika testing kurang memuaskan maka

akan dikerjakan apa-apa saja yang kurang lalu testing lagi. Model ini cocok pada proyek dengan fokus pada kualitas tinggi

- Prototype model: model ini berfokus pada pengembangan prototype dan pemahaman pengguna pada perangkat lunak maka pada step-by-step nya juga prototype dan user oriented. Tahapannya berurutan dari permintaan - desain singkat - pembuatan prototype - evaluasi user - refining prototype - implementasi dan maintain.
- Spiral model: model ini menggabungkan elemen model spiral dengan pendekatan inkremental jadi setiap siklusnya akan ada penambahan fitur. Model ini cocok untuk proyek besar berisiko dan kompleks
- Iterative incremental model: model ini seperti model spiral tetapi banyak pengulangan pada analysis dan design hingga lalu code dan test setelah selesai sekali akan dilakukan pada fitur lainnya. Jadi ini dilakukan untuk menambah fitur kecil-kecilan pada perangkat lunak
- Bigbang model: model yang dilakukan tanpa tahapan
- Agile model: pendekatan kolaborative yang dinamis dan terbuka untuk berbagai perubahan pada setiap tahapan

b. Git

Git adalah kontrol versi terdistribusi sehingga pengguna dapat mengetahui perkembangan perangkat lunak, kolaborasi kode atau revisi kode

Git terdiri dari git init, git clone, git status dan git add. Git commit, git push, git pull, git branch git checkout, git merge, git log, git remote, git fetch, git diff dan git reset. Selengkapnya akan dijelaskan berikut ini:

1. Otentikasi akun terlebih dahulu

```
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git push origin ufairah-bashir
info: please complete authentication in your browser...
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 12 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 3.05 MiB | 1.02 MiB/s, done.
Total 4 (delta 1), reused 1 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'ufairah-bashir' on GitHub by visiting:
remote:   https://github.com/ikrar557/it-perbankan-finpro-kelompok3/pull/new/ufairah-bashir
remote:
To https://github.com/ikrar557/it-perbankan-finpro-kelompok3
 * [new branch]      ufairah-bashir -> ufairah-bashir
```

2. Clone git repositeri di website

```
Please make sure you have the correct access rights
and the repository exists.
PS C:\Users\ufairah bashir> git clone https://github.com/ikrar557/it-perbankan-finpro-kelompok3
```

3. Buka file git di laptop masing-masing dengan cd (nama file)

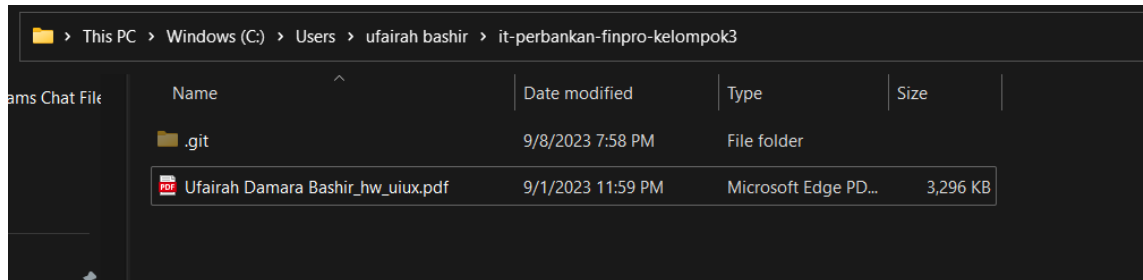
```
PS C:\Users\ufairah bashir> cd it-perbankan-finpro-kelompok3
```

4. Buat brach yaitu cabang file ini seperti hanya terminal saja yang menganggap branch ini ada. Terminal yang dibuka saat ini. Misalkan

membuat branch hello-kitty pada tab terminal yang dibuka, itulah branch nya. Jadi kita buat branch dengan cara : git checkout -b ufairah-homework

```
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git checkout -b ufairah-homework
Switched to a new branch 'ufairah-homework'
```

5. Sekarang buka file explorer, buka alamat yang dibuka git (C:\Users\ufairah bashir\.....\nama repo yang di clone) buat/masukkan/hapus file ke/di folder tersebut



6. Sudah itu ke terminal lagi, cek apakah terjadi perubahan di folder yang di clone berdasarkan branch yang dibuat (di sini terjadi karena ada tulisan berwarna merah. Tertulis delete karena saya telah menghapus file yang tertulis judulnya yang bertuliskan merah itu dan menyisakan file di nomor 5

```
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git status
On branch ufairah-homework
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    2_UIUX_Ufairah Damara bashir.pdf
```

7. Git add . adalah file tersebut ditambahkan lalu lakukan commit untuk memberikan note mengenai file apa yang ditambahkan itu. File yang git add . belum masuk ke github repo, masih di laptop (ini contoh untuk branch ufairah-bashir karena branch ufairah-homework tidak ke step git status langsung ke step git push origin)

```
nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git add .
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git status
On branch ufairah-bashir
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   Ufairah Damara Bashir_hw_uiux.pdf
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git commit -m "rename homework uiux ufairah"
```

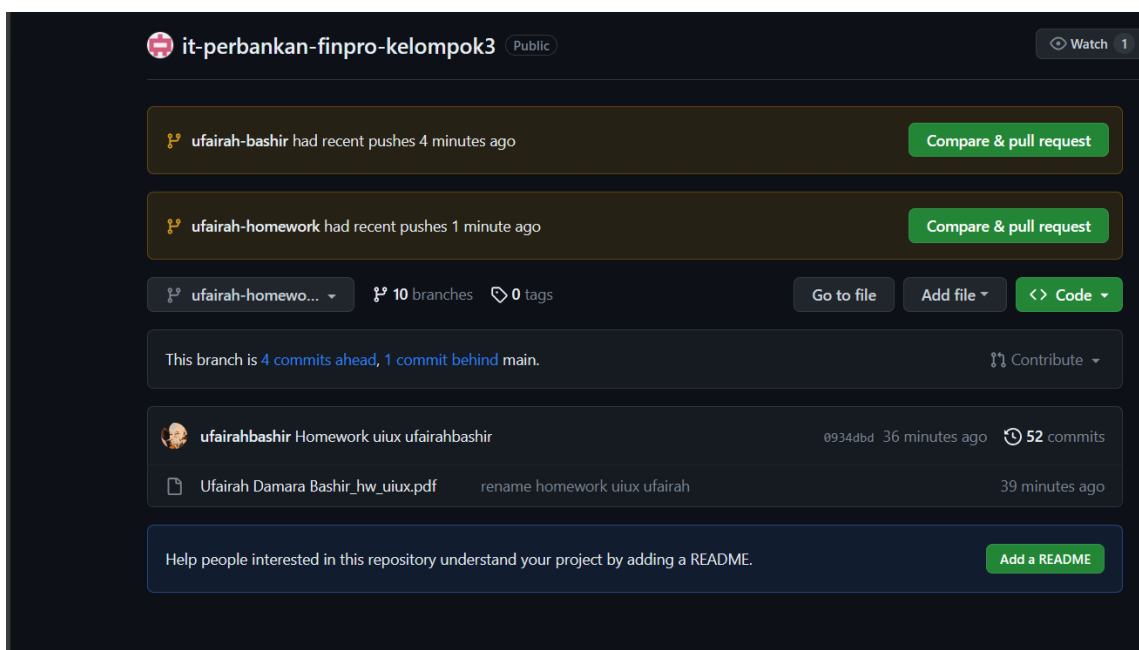
8. Git push origin ini akan update di github yang sebenarnya.

```

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git add .
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git commit -m "Homework uiux ufairahbashir"
[ufairah-homework 0934dbd] Homework uiux ufairahbashir
1 file changed, 30006 deletions(-)
delete mode 100644 2_UIUX_Ufairah Damara bashir.pdf
PS C:\Users\ufairah bashir\it-perbankan-finpro-kelompok3> git push origin ufairah-homework
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 258 bytes | 258.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'ufairah-homework' on GitHub by visiting:
remote:   https://github.com/ikrar557/it-perbankan-finpro-kelompok3/pull/new/ufairah-homework
remote:
To https://github.com/ikrar557/it-perbankan-finpro-kelompok3
 * [new branch]      ufairah-homework -> ufairah-homework

```

9. done!!



- c. Full Stack Web Development
Web development menguasai bahasa pemrograman HTML, CSS, Javascript. Full stack dev menggunakan framework react, vue.js dan angular js. Selain itu belajar juga back end development dan database development. Mulai dari antar muka dibuat dengan kode sampai hosting database.
- d. Front end Engineering
Front membangun antar muka pengguna yang menarik menggunakan html, css dan java script. Sama seperti full stack web development tetapi tidak membangun kode untuk back end dan database nya. Hanya berfokus pada pembuatan wadah antar muka dengan kode yang tadi dijelaskan.
- e. Back end Engineering
Back end engineering berfokus pada otak aplikasi yang menyambungkan database dengan front end. Bahasa pemrograman yang digunakan

adalah server side seperti python, java, php, C# dan lain lain. Selain itu server framework yang digunakan tergantung bahasa yang digunakan. Misalkan laravel untuk php atau flask untuk python. Lalu mengerjakan database management seperti sql (my sql, postgresql, sql server) dan no sql (mongo db dan firebase)

f. Mobile Engineering

Berbeda dari web engineering mobile dikembangkan untuk platform seperti android, ios atau windows phone. Bahasa pemrograman bergantung pada platform yang ditargetkan misalkan android menggunakan java/kotlin dan iOS menggunakan swift/objective-C

g. collaboration

Kolaborasi efektif dengan version control misalkan git.

Manfaat dari version control adalah:

- Rekam perubahan
- Pencatatan riwayat
- Pemecahan konflik
- Pemulihan mudah

Fungsi version control untuk kolaborasi:

- Inisiasi proyek: memulai proyek, membuah repositori. Anggap saja seperti lemari kosong yang akan ditempati lembaran kode atau lainnya
- Pengembangan paralel: tiap anggota tim mempunyai salinan repositori pada komputer masing-masing. Anggap saja seperti setiap anggota kelompok mempunyai lemari masing-masing yang akan digabungkan di pusat lemari.
- Branching: branchings seperti file file dalam lemari. File ini bermanfaat agar jika satu file rusak, masih ada file lain sejenis yang masih terseleamatkan atau jika satu file rusak, tidak merusak file lainnya.
- Merge: menyatukan semua cabang ke dalam satu cabang utama. Jadi file-file kecil tadi akan dimasukkan ke file utama jika sudah tidak ada lagi yang perlu dirubah.
- Pull request: ini seperti mengajukan file kita apakah sudah tepat ke pemilik lemari/file pusat. Jika sudah tepat, pemilik file pusat/lemari akan menerima file kecil kita untuk dimasukkan ke file pusat.