

# Git Sebagai Media Pembelajaran

## Mata Kuliah Pemrograman

Oleh : M. Saiful Mukharom – Komunitas CAH UNP

### 1. Pendahuluan

#### 1. 1. Latar Belakang

Dalam pengembangan *software*, *Git* adalah sebuah *Distributed Revision Control* (DRC) dan *Source Code Management* (SCM) System dengan penekanan pada kecepatan (Linus, Torvalds, 2005). *Git* merupakan sistem pengontrol versi terdistribusi yang populer untuk proyek – proyek *open source*<sup>1</sup>. *Git* bisa sebagai media pembelajaran terbaik untuk berbagi kode dengan teman-teman sekelas, melakukan hal besar dengan cara belajar bersama<sup>2</sup>. Sebuah layanan hosting yang menggunakan sistem pengontrol versi *git* yaitu *GitHub*, pada bulan Februari 2014 telah resmi meluncurkan layanan untuk pendidikan<sup>3</sup>, sebuah situs yang di dedikasikan untuk menggunakan *GitHub* di lingkungan pendidikan. Konsep *social coding* perlu diterapkan pada pembelajaran mata kuliah pemrograman, karena pendekatan ini mengarah pada pembelajaran kooperatif<sup>4</sup>, sehingga perkembangan pemahaman mahasiswa tentang pemrograman lebih cepat dan optimal.

#### 1. 2. Tujuan dan Manfaat

- Membantu mempermudah mahasiswa dalam memahami mata kuliah pemrograman.
- Menerapkan pembelajaran kooperatif dengan konsep *social coding*.
- Memudahkan mahasiswa untuk mengetahui pekerjaan pemrograman mahasiswa lain.

### 2. Pembahasan

Tahapan ini akan melakukan simulasi tentang bagaimana dosen mata kuliah pemrograman C++ memberikan tugas kelompok kepada mahasiswa dengan memanfaatkan *Sourceforce GitHub* sebagai media pembelajarannya, kemudian mahasiswa melakukan kolaborasi dengan mahasiswa lain untuk menyelesaikan tugasnya dan terakhir yaitu dosen melakukan pemeriksaan hasil tugas mahasiswa yang telah diberikan.

#### 2.1. Dosen Memberikan Tugas

##### a. Mempersiapkan akun *GitHub*.

Pada tahapan ini dosen maupun mahasiswa melakukan registrasi akun pribadi di *GitHub*<sup>5</sup>.

##### b. Membuat organisasi baru bernama *Class-Room*.

Anda harus masuk sebagai akun *GitHub* pribadi untuk membuat akun organisasi. Tahapan ini adalah untuk mengorganisir tugas mahasiswa.



Ada tiga tahapan dalam membuat organisasi baru yaitu mempersiapkan akun nama organisasi apa yang akan digunakan misalnya *Class-Room*, menambahkan *owner* hak akses organisasi misalnya menambahkan asisten dosen, kemudian menambahkan mahasiswa sebagai *member* organisasi *Class-Room*.

A screenshot of the GitHub interface showing a button labeled 'Add a member by username' with a person icon.

Untuk mendapatkan *private repository* bisa melakukan *request discount*<sup>6</sup> sesuai

<sup>1</sup> <https://git-scm.com>

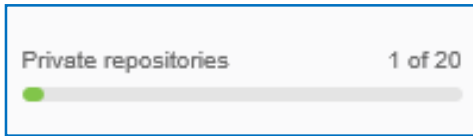
<sup>2</sup> <https://github.com/about>

<sup>3</sup> <https://education.github.com/>

<sup>4</sup> <http://blog.elearning.unesa.ac.id/alim-sumarno/model-pembelajaran-kooperatif>

<sup>5</sup> <https://help.github.com/articles/signing-up-for-a-new-github-account>

kebutuhan misalnya Sembilan *private repository*, apabila berhasil mendapatkan permohonan discount maka akan ditunjukkan seperti.



- c. Membuat silabus tugas.

Pada tahapan ini yaitu membuat deskripsi tugas<sup>7</sup> untuk mempermudah dan memperjelas perintah apa yang akan dilakukan oleh mahasiswa.

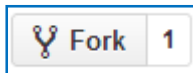
- d. Membuat repositori tugas<sup>8</sup>.

Mempersiapkan repositori untuk memberikan tugas kepada mahasiswa semisal membuat bahasa pemrograman C++ persegi panjang<sup>9</sup>.

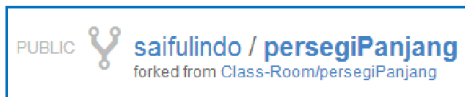
## 2.2. Mahasiswa Mengerjakan Tugasnya.

- a. Mahasiswa melakukan *forking*.

Setelah dosen memberikan tugasnya kemudian mahasiswa melakukan *forking* untuk menggandakan ke akun pribadinya.



Kemudian akan terlihat seperti :



- b. Mahasiswa melakukan *clone*.

Sebelum melangkah ketahapan ini perlu melakukan instalasi *Git* pada *windows* maupun *linux* sesuai sistem operasi yang digunakan<sup>10</sup>.

Mahasiswa melakukan *cloning* dengan perintah `$ git clone`

```
git://github.com:saifulindo/persegiPanjang.git
```

```
$ git clone git://github.com:saifulindo/persegiPanjang.git
Cloning into 'persegiPanjang'...
remote: Counting objects: 17, done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 17 (delta 4), reused 17 (delta 4)
Receiving objects: 100% (17/17), 270.93 KiB | 43.00 KiB/s, done.
Resolving deltas: 100% (4/4), done.
Checking connectivity... done.
```

- c. Mahasiswa mengerjakan tugas.

Pada tahapan ini melakukan *editing* tugas yang telah diberikan dengan cara seperti berikut :

```
$ cd persegiPanjang/
```

```
$ touch persegiPanjang.cpp
```

Modifikasi file `persegiPanjang.cpp` :

```
$ notepad persegiPanjang.cpp
```

```
#include <iostream>
using namespace std;

int main()
{
    double panjang, lebar;

    panjang = 20;
    lebar = 10;

    keliling = 2 * panjang + lebar;
    Luas = panjang*lebar;

    cout << "Keliling persegi panjang = "
         << keliling << endl;
    cout << "Luas Persegi Panjang = "
         << Luas << endl;

    return 0;
}
```

Kemudian melakukan *staging index* dan *commit* untuk di rekam dan disimpan ke *git repository*.

```
$ git add .
```

```
$ git commit -m "menambahkan
persegiPanjang.cpp dan menambahkan script
menampilkan luas"
```

- d. Mengumpulkan tugas.

Untuk mengumpulkan tugas dengan cara melakukan pull dan push.

```
$ git pull origin master
```

```
$ git pull origin master
From github.com:saifulindo/persegiPanjang
 * branch                master       -> FETCH_HEAD
 * [new branch]          master       -> origin/master
Already up-to-date.
```

```
$ git push origin master
```

```
$ git push origin master
Counting objects: 5, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 310 bytes | 0 bytes/s, done.
Total 3 (delta 2), reused 0 (delta 0)
To git://github.com:saifulindo/persegiPanjang.git
95d1645..4c03059 master -> master
```

Kemudian melakukan *create pull request*.



Kemudian ketikkan pesan misalny "mengirim tugas", kemudian Klik [*Create pull request*].

- 2.3. Dosen Memeriksa Tugas Mahasiswa.

Memeriksa dengan cara melihat *pull request* :

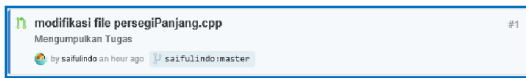
<sup>6</sup> [https://education.github.com/discount\\_requests/new](https://education.github.com/discount_requests/new)

<sup>7</sup> <https://github.com/advanced-js/syllabus>

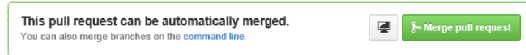
<sup>8</sup> <https://help.github.com/articles/create-a-repo>

<sup>9</sup> <https://github.com/Class-Room/persegiPanjang.git>

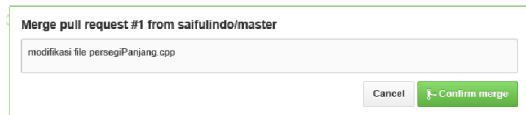
<sup>10</sup> <http://cah.unpkediri.ac.id/instalasi-git-di-windows-dan-linux/>



Kemudian dosen bisa mengklarifikasi atau memberikan masukan bila dibutuhkan, kemudian apabila dirasa sudah sesuai tugasnya, maka dosen melakukan *merge pull request* :

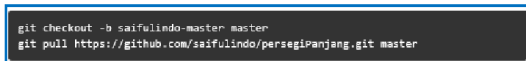


Kemudian *confirm merge* :

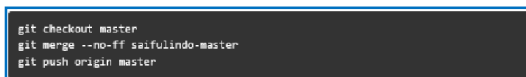


Atau bisa melakukan penggabungan (*merge*) via *command line* :

Langkah pertama, dari repositori proyek Anda, periksa cabang baru dan menguji perubahan.



Langkah kedua, Menggabungkan perubahan dan *update* pada *GitHub*.



### 3. Kesimpulan

*Git* sangat sederhana dan mudah dalam pengoperasiannya serta sangat sesuai jika diterapkan didunia pendidikan pada mata kuliah pemrograman, untuk tugas kelompok dan melakukan hal besar dengan teman sekelas untuk mengembangkan *source code*. Berdasarkan pendahuluan dan pembahasan bahwa dapat disimpulkan beberapa hal (1) Antar mahasiswa dapat saling melihat dan duplikasi tugas sehingga membuka wawasan mahasiswa lebih luas tentang pemrograman. (2) Dosen dan mahasiswa dapat terus memperbaiki tugas yang diberikan maupun yang dikerjakan, kemudian mahasiswa adhik tingkat dapat mengikuti perkembangan pemrograman, sehingga lebih cepat penguasaanya tentang pemrograman ketika mendapatkan mata kuliah tersebut. (3) dan pada akhirnya mata kuliah pemrograman terus berkembang dan terus mengalami perbaikan kualitas.

### 4. Daftar Pustaka

- [1] Sumarno , Alim, M.Pd, 2011. Model Pembelajaran Kooperatif. (online).tersedia: <http://blog.elearning.unesa.ac.id/alim-sumarno/model-pembelajaran-kooperatif>. diunduh januari 2014.
- [2] Education, GitHub, 2014. Classroom Guide. (online). Tersedia : <https://education.github.com/guide>. diunduh Juni 2014
- [3] John, Britton. 2014. GitHub Goes to School. (online). Tersedia : <https://github.com/blog/1775-github-goes-to-school>. diunduh Juni 2014.
- [4] Orsini, Lauren. 2014. GitHub Gets Its Science On. (online). Tersedia : <http://readwrite.com/2014/05/15/github-science-doi-academia-repository>. diunduh Juni 2014.

### 5. Perpustakaan

Untuk memahami apa itu *Git*, *GitHub*, *Repository*, *Clone*, *Branch* dan yang lainnya anda bisa melihat *GitHub Glossary*<sup>11</sup>.

<sup>11</sup> <https://help.github.com/articles/github-glossary>