**LAPORAN PRAKTIKUM**

**PEMROGRAMAN BERBASIS WEB**

**“PRATIKUM 1”**



Disusun Oleh :

Nama : Lalu Muhamad Waisul Kuroni

Nim : 1808561037

Kelas : B

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS UDAYANA**

**BADUNG**

**2020**

# **KATA PENGANTAR**

Puji dan syukur ke hadirat Tuhan Yang Maha Kuasa atas segala rahmat yang diberikan-Nya sehingga tugas Laporan Praktikum 1 yang berjudul ”Pertemuan 1” ini dapat saya selesaikan. Laporan ini saya buat sebagai kewajiban untuk memenuhi tugas. Dalam kesempatan ini, penulis menghaturkan terimakasih yang dalam kepada semua pihak yang telah membantu menyumbangkan ide dan pikiran mereka demi terwujudnya makalah ini. Akhirnya saran dan kritik pembaca yang dimaksud untuk mewujudkan kesempurnaan makalah ini penulis sangat hargai.

Pujut, 29 September 2020

Penyusun

# **DAFTAR ISI**

[**KATA PENGANTAR** i](#_Toc52302424)

[**DAFTAR ISI** ii](#_Toc52302425)

[**BAB I PENDAHULUAN** 1](#_Toc52302426)

[**1.1** **Latar Belakang** 1](#_Toc52302427)

[**1.2** **Tujuan** 1](#_Toc52302428)

[**1.3** **Manfaat** 1](#_Toc52302429)

[**BAB II KAJIAN PUSTAKA** 2](#_Toc52302430)

[**2.1 Pengertian Github** 2](#_Toc52302431)

[**2.2 Fungsi Github** 2](#_Toc52302432)

[**BAB III PEMBAHASAN** 6](#_Toc52302433)

[**3.1 Permasalahan** 6](#_Toc52302434)

[**3.2 Penyelesaian** 6](#_Toc52302435)

[**BAB IV PENUTUP** 9](#_Toc52302436)

[**4.1. Simpulan** 9](#_Toc52302437)

[**4.2. Saran** 9](#_Toc52302438)

# **BAB I PENDAHULUAN**

* 1. **Latar Belakang**

Dalam bekerja tim pasti akan terjadi perselisihan yang akan memberikan dampak buruk dalam sebuah proyek jika tidak dapat diselesaikan dengan baik, untuk menyelesaikan hal tersebut tetunya diperlukan kerja sama tim yang baik oleh karena itu dibutuhkan wadah untuk menyelesaikan proyek tersebut dengan berkolaborasi dengan baik. Saaat ini terdapat berbagai macam wadah untuk menyelesaikan masalah tersebut salah satunya adalah GitHub.

Git adalah control software yang terdistribusi. Software control adalah cara untuk menyimpan perubahan dari waktu ke waktu tanpa menimpa versi sebelumnya. Didistribusikan berarti bahwa setiap pengembang yang bekerja dengan repositori Git memiliki salinan dari seluruh repositori itu - setiap komit, setiap cabang, dan setiap file. Maka untuk itu laporanini akan mengupas tentang Penggunaan GitHub serta mempraktikan lansung dengan cara membuat konten dengan file xhtml.

* 1. **Tujuan**
     + Memenuhi tugas Pratikum Pemrograman Berbasis Web.
     + Bagaimana penggunaan GitHub.
     + Bagaimana proses pembuatan konten dengan file xhtml.
  2. **Manfaat**
     + Untuk memenuhi tugas Praktikum Pemrograman Berbasis Web.
     + Untuk memahami tentang penggunaan GitHub.
     + Untuk dapat membuat konten dengan file xhtml.

# **BAB II KAJIAN PUSTAKA**

## **2.1 Pengertian Github**

Git adalah control software yang terdistribusi. Software control adalah cara untuk menyimpan perubahan dari waktu ke waktu tanpa menimpa versi sebelumnya. Didistribusikan berarti bahwa setiap pengembang yang bekerja dengan repositori Git memiliki salinan dari seluruh repositori itu - setiap komit, setiap cabang, dan setiap file.

## **2.2 Fungsi Github**

GitHub berfungsi untuk membantu penyimpanan repository. Namun tak hanya itu, dengan aplikasi ini pengguna juga dapat melakukan kolaborasi dalam mengerjakan project tertentu agar dapat terus melakukan update secara rutin, termasuk melacak dan menyimpan perubahan-perubahan yang terjadi dalam project pengguna tersebut. Aplikasi ini tidak hanya dapat digunakan oleh para programmer atau orang-orang yang sedang memiliki project saja. Aplikasi ini dapat digunakan secara luas oleh siapa saja, termasuk oleh Pengguna yang saat ini sedang mencari pekerjaan. Aplikasi ini memiliki laman profil yang akan memuat data probadi Pengguna, seperti foto, email, bahkan followers yang Pengguna miliki. Informasi-informasi yang dimuat dalam profil pengguna ini akan menarik pihak lain untuk mengajak pengguna bekerjasama dalam sebuah project.

Berikut ini beberapa fungsi dari GitHub:

* Dapat digunakan untuk menyimpan repository.
* Menjadi wadah atau alat untuk kolaborasi dan mengerjakan sebuah project.
* Memuat profil Pengguna sehingga dapat mengikuti programmer lainnya di aplikasi ini.
* Dapat untuk mengawasi repository tertentu.

2.3 Fitur – fitur pada Github

Adapun fitur – fitur pada GitHub seperti berikut:

1. Fitur pencarian

Pada fitur pencarian ini Pengguna dapat melakukan pencarian hal yang ingin Pengguna cari. Karena Github sendiri mempunyai 10 juta lebih software, maka Pengguna dapat mencarinya pada fitur pencarian dengan menuliskan kode nya. Nah, pada Github ini hasil pencarian akan difilter sesuai dengan repo, kode, issue, dll. Selain itu pencarian yang dilakukan pada Github juga bisa diurutkan sesuai dengan jumlah bintang atau sesuai dengan jumlah “fork”. Jadi Pengguna langsung bisa menggunakan fitur pencarian ini dengan cara mengetik kode yang ingin Pengguna cari pada kolom pencarian.

1. Clone

Didalam Github terdapat dua kode yaitu repository lokal dan repository server. pada clone ini Pengguna dapat melakukan duplikasi terhadap kode pada repository server dan dipindah pada repository lokal. Nah, pada repository server Pengguna bisa membuka http/https dan ssh. Perintah yang terdapat pada Github adalah menggunakan: git clone <URL Repository> <opsional: nama folder>.

1. Membuat/mengambil/mengatur/mengembangkan repository

Seperti yang sudah dijelaskan diatas bahwa repository merupakan hal yang sangat penting dalam Github. Untuk itu Pengguna harus mengubah repository kedalam Github yaitu dengan cara mengoperasikan perintah git init <directory>. Nah, setelah Pengguna mengetik dan mengoperasikan hal tersebut, maka akan muncul folder yang bernama .gt pada direktori yang kalian buat. Namun, jika sebelumnya Pengguna sudah memiliki direktori dan berkeinginan untuk mengambilnya maka kalian tinggal klik pada perintah git clone. Terdapat perbedaan pengoperasian antara direktori lokal dan direktori yang sudah tersimpan sebelumnya. Untuk pengoperasian direktori lokal Pengguna bisa menggunakan git clone/path/to/local/repository. Namun jika Pengguna ingin mengambil repositori yang sudah tersimpan sebelumnya maka Pengguna bisa menggunakan git clone user. name@host: /path/to/remote/repository. lebih mudah lagi jika Pengguna memiliki akun Hostinger, maka Pengguna dengan mudah langsung dapat menyalin, mengatur, dan mengembangkan repository melalui Member Area -> GIT. Misalnya jika Pengguna menyalin repository pada Github maka Pengguna tinggal memasukkan alamatnya, lalu memilih branch dan lokasi install kemudian klik tombolnya.

1. Branches

Melalui fitur branch ini Pengguna bisa membuat branch independen yang berbeda dalam satu project yang Pengguna buat. Apa sih branch itu? Ya, branch adalah fitur yang ada pada Github untuk membuat fitur-fitur yang keren namun tidak terlihat oleh yang lain. Jika Pengguna ingin membuat fitur pada branch maka Pengguna bisa menggunakan git checkout –b nama branch n \*. Dan jika Pengguna ingin ingin kembali ke master utama maka Pengguna bisa mengoperasikan hal ini git checkout master. Selain itu jika Pengguna ingin menghapus sebuah branch maka Pengguna bisa menggunakan git checkout –b nama branch n. Selanjutnya apakah Pengguna ingin membuat branch yang bisa digunakan oleh pengguna lainnya, maka Pengguna harus mengirimnya ke remote repository dengan menggunakan git push origin nama branch n.

1. Add & Commit

Pada fitur ini Pengguna bisa mengajukan perubahan, mengajukan tambahan yang akan ditambahkan pada index dengan menjalankan perintah add. Lalu bagaimana caranya? Caranya adalah dengan menggunakan git add <file\_name>. Namun, jika Pengguna sudah yakin dengan perubahan yang akan Pengguna lakukan maka langsung saja mengadakan perubahan pada HEAD dan langsung mengikuti perintah dari commit yaitu dengan menggunakan git commit –m “Masukkan catatan untuk commit kalian”. Perlu Pengguna ketahui bahwa saat perintah commit dijalankan, file belum terkirim ke remote repository namun file akan langsung di commit ke HEAD. Artikel terkait Cara Memulai Bisnis Online untuk Bisa Sukses

1. Mengirim perubahan melalui Github

Nah, selanjutnya jika Pengguna sudah yakin akan melakukan perubahan dan sudah di commit ke HEAD maka langsung saja file tersebut Pengguna kirim ke repository dengan menggunakan git push origin master yang dioperasikan melalui direktori kerja dan perubahan akan langsung dikirim pada remote repository. Keunggulannya adalah bahwa master dapat Pengguna ganti dengan nama branch sesuai dengan perubahan yang Pengguna inginkan pada commit. Namun jika repository belum Pengguna clone, tetapi Pengguna ingin menyambungkannya dengan repository dan remote server, maka Pengguna bisa menggunakan git remote add origin <server>. Nah, setelah itu perubahan yang sudah dibuat dan sesuai dengan keinginan Pengguna akan langsung dikirim ke server.

1. Push

Pada fitur ini Pengguna bisa merubah kode yang didapatkan dari commit pada repository lokal dapat di push up pada server umum. Agar terjadi antara pengembang lain untuk menambah wawasan dan pengetahuan antar pengguna yaitu dengan menggunakan: git push <nama remote repository> <nama branch>.

1. Pull

Pada fitur ini Pengguna bisa mendownload kode yang terdapat pada repository server menuju repository lokal atau milik pribadi. Pengguna bisa melakukan pull jika semua kode sudah di commit terlebih dahulu atau bisa juga digunakan jika memang tidak ada perubahan kode. Pengguna bisa melakukan pull dengan menggunakan: git pull <nama remote repository> <nama branch>.

1. Update & Merging

Melalui fitur ini Pengguna bisa memperbarui / mengupdate direktori kerja lokal Pengguna menjadi bentuk yang paling baru. Untuk melakukan hal tersebut Pengguna bisa menggunakan git pull. Selain itu Pengguna dapat menggabungkan branch lain kedalam branch yang sudah aktif yaitu dengan menggunakan git merge nama branch n. Keunggulan fitur update & merging ini adalah bahwa Pengguna tidak perlu susah-susah menangani masalah yang timbul karena dengan adanya fitur ini, Github akan selalu menangani masalahnya sendiri, namun terkadang juga mengalami kegagalan. Nah, lalu bagaimana jika mengalami kegagalan? Pengguna tidak perlu khawatir karena penanganannya sangat mudah yaitu Pengguna langsung saja bisa melakukan resolve conflict atau menyelesaikan masalah secara manual. Setelah itu Pengguna bisa mengedit file untuk menghapus permasalahan yang timbul dan langsung Pengguna tpenggunai merge dengan menggunakan git add <nama\_file>. Apakah Pengguna penasaran dengan perubahan yang terjadi? Jika iya, maka Pengguna bisa melihatnya kembali dengan menggunakan git diff <nama\_sumber\_branch> <nama\_target\_branch>.

1. Tagging

Fitur tagging ini dapat Pengguna lakukan sebelum Pengguna membuat perubahan, mengatur, menyalin, bahkan mengupdate apapun. Karena tagging sangat diperlukan sebelum melakukan berbagai hal tersebut. Pengguna bisa melakukan tagging dengan menggunakan git tag 1. 1 .0 sepuluh karakter pertama dari commit-id, contohnya adalah git tag 1. 1 .0 1a2bd3d67fb. Lalu dimana Pengguna bisa mendapatkan sepuluh karakter commit-id tersebut? Nah, Pengguna bisa mendapatkan sepuluh karakter tersebut pada log.

1. Log

Apa itu log? Ya, dalam fitur ini Pengguna bisa melihat catatan history yang pernah Pengguna buat pada repository. Semua catatan history pada repository dapat dilihat dari log. Nah, untuk melihat commit yang pernah dibuat oleh pengguna lain, kalian bisa menggunakan: git log –author =nama pengguna, contohnya adalah: git log –author =Rina. Lalu untuk melihat 1 commit menjadi per baris atau log yang sudah terkompresi maka Pengguna bisa menggunakan: git log –rina=online. Sedangkan log yang digunakan untuk melihat yang berubah adalah dengan menggunakan: git log –name-status. Kemudian untuk membuat perubahan di lokal karena mungkin Pengguna tidak sengaja melakukan kesalahan dan ingin kembali pada perubahan sebelumnya, maka Pengguna bisa menggunakan fitur log yaitu dengan menggunakan: git checkout — <filename>. Nah, jika Pengguna ingin melihat history pada perubahan di drop dan master branch lokal maka Pengguna membutuhkan history terakhir dari server tersebut, maka Pengguna bisa menggunakan: git fetch origin. Nah, itulah beberapa fitur dan penggunaannya pada Github yang sangat canggih dan mudah. Selain fitur-fitur tersebut, keunggulan Github dibanding dengan project hosting yang lain akan dibahas dibawah ini.

# **BAB III PEMBAHASAN**

## **3.1 Permasalahan**

Buatlah sebuah laporan tentang bagaimana cara membuat directory dengan nama Praktikum 1 dan cara mengupload file ke directory tersebut pada github. File yang di upload merupakan file xhtml dan laporan yang kalian kerjakan. Untuk konten yang dibuat pada file xhtml adalah menampilkan nama dan nim kalian!

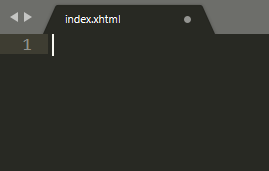
## **3.2 Penyelesaian**

mempersiapkan hal berikut:

* Browser
* Text Editor
* Akun GitHub

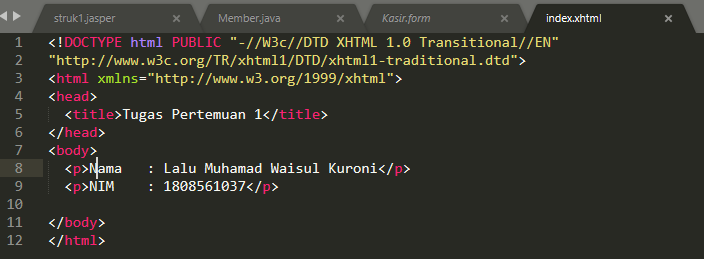
Jika sudah, maka berikut ini langkah-langkah penyelesaian masalah:

1. Buka text editor dan simpan file dengan nama index.xhtml



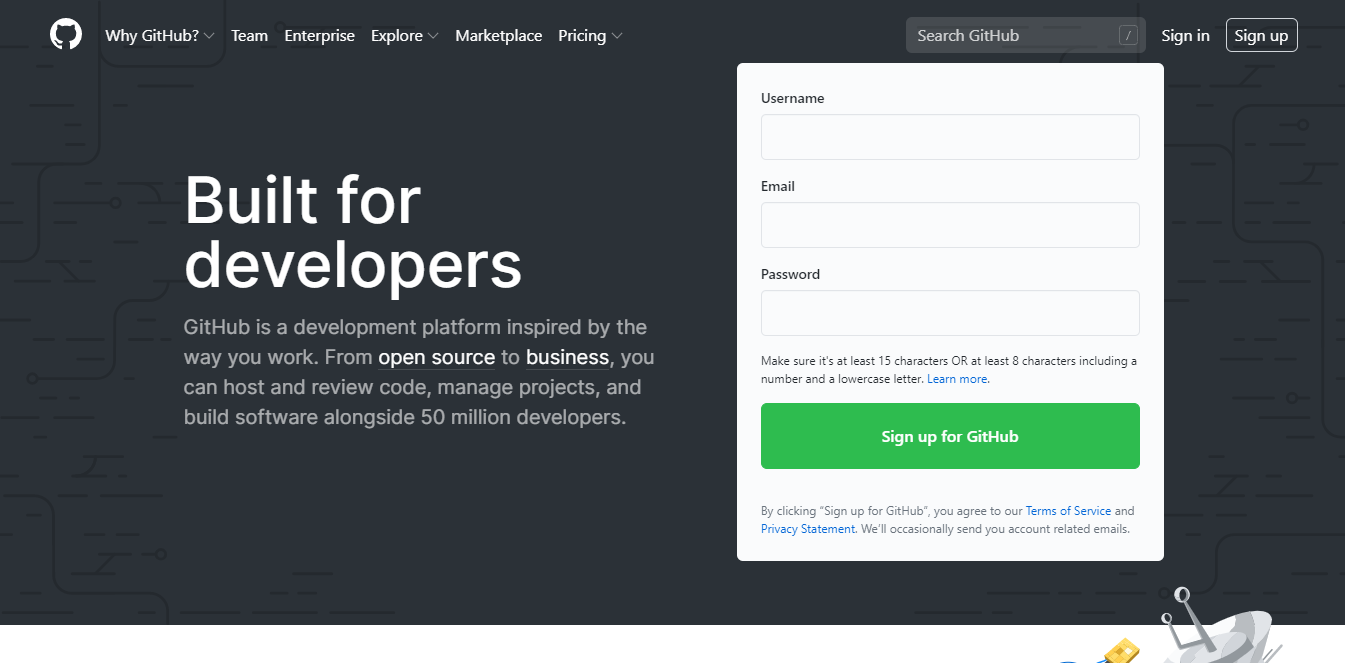
Penyimpanan pada sublime text dapat menggunakan shortcut ctrl+s dan ketik nama file beserta ekstensi .xhtml.

1. Ketik source code berdasarkan tugas pratikum 1



Deklarasi Doctype pada code di atas menggunakan tipe tradisional yang digunakan untuk membuat halaman yang sebagian besar menggunakan tag-tag HTML. Setelah itu, dilanjutkan dengan kode-kode element untuk pembuatan konten.

1. Buka browser dan masuk ke dalam website github (<https://github.com/>) dan login.



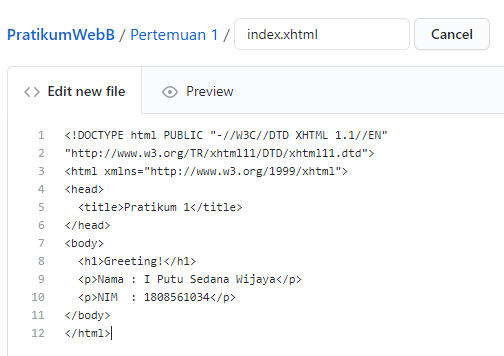
1. Masuk ke dalam repository PratikumWebB



1. Membuat folder dan membuat file xhtml di dalam repository *PraktikumWebB*

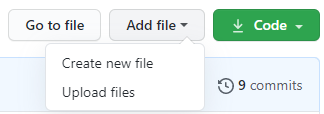


Pilihlah Create new file Untuk membuat folder dan tulis nama folder / nama fil xhtmlpada kolom seperti pada gambar dan tekan tombol commit di bawah.

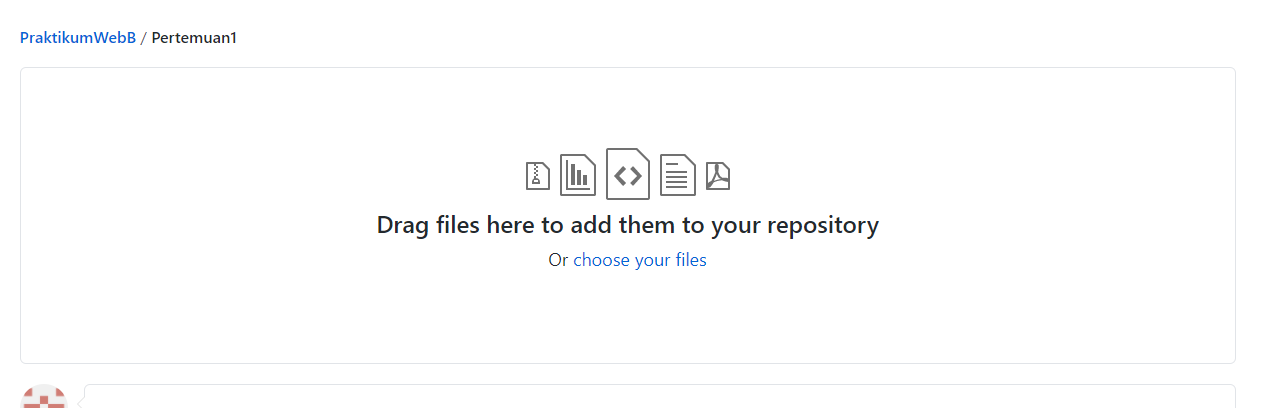


Selanjutnya tambahkan code yang telah dibuat pada text editor ke dalam kolom text GitHub dan tekan kembali tombool commit yang ada di bawah

1. Selanjutnya mengupload file laporan



Masukan ke dalam folder “Pertemuan 1” pada repository PraktikumWebB. Lalu lalu pilih upload files



Upload file dapat dilakukan dengan drag file yang ingin diupload ke dalam kotak yang telah di sediakan. Setelah file terupload tekan lah tombol commit yang ada di bawah.

# 

# **BAB IV PENUTUP**

## **4.1. Simpulan**

Git adalah control software yang terdistribusi yang akan membantu proses kerjasama tim yang berperan sebagai wadah yang menyimpan salinan dari seluruh repositori untuk setiap komit, setiap cabang, dan setiap file.

## **4.2. Saran**

Dengan memahami cara penggunaan GitHub diatas hendaknya kita mampu memahami serta mempraktekkannya dalam melakukan kerjasama tim untuk mencapai tujuan yang diinginkan.

**DAFTAR PUSTAKA**

GitHub, *Git Guide*, dilihat 29 September 2020, < <https://github.com/git-guides>>

Andy, 2020. *Mengenal Apa Itu GitHub dan Cara Menggunakannya*, Qwords, dilihat 29 September 2020, <[https://qwords.com/blog/apa-itu-github/](https://qwords.com/blog/apa-itu-github/#:~:text=Secara%20lebih%20sederhana%2C%20beberapa%20fungsi,programmer%20lainnya%20di%20aplikasi%20ini).>