HOW TO MAKE USE GITHUB DESKTOP FOR DEVELOPING WEBSITE PROJECT

This report was written to show the author responsibility while doing Field Work Practice program at

PT. DUTA COMPUTER

Period January 7th, 2019 until May 7th, 2019

MUHAMMAD KHALIL ALFARIZI

NIS: 17.7724

Skill Program : **Information and Communication**

Skill Competency : **Computer and Network Engineering**

RIAU ISLAND EDUCATIONAL DEPARTMENT

VOCATIONAL HIGH SCHOOL 1 BATAM

Jl. Prof. Dr. Hamka 1 Batu Aji – Batam. Phone (0778) 365903

2019

MOTTO

Hard work will never betray you.

INDUSTRY APPROVAL PAGE

HOW TO MAKE USE GITHUB DESKTOP FOR DEVELOPING WEBSITE PROJECT

This report has approved and authorized as one of the field work practice programs at the 4th semester in **Vocational High School 1 Batam**

This approval sheet has been authorized in

PT. DUTA COMPUTER

On Tuesday, May 7th 2019 by:

Industry Advisor,	**Intern Coordinator,**
RUSLAN	**DEKCY**
Chief Technician	Mentor

PT. DUTA COMPUTER

SCHOOL APPROVAL PAGE

HOW TO MAKE USE GITHUB DESKTOP FOR DEVELOPING WEBSITE PROJECT

This report has approved and authorized as one of the practical working requirement at the 4th semester in **Vocational High School 1 Batam**
This approval sheet has been authorized in
Vocational High School 1 Batam
On Saturday May 7th 2019 by:

Industrial Coordinator,	**Counselor of Report,**
Junaidi, S.Pd	**Yarliasyah Agustian, S.ST**
NIP. 19820711 200903 1 003	NIP.19860802 200903 1 002

Headmaster of Vocational High School 1 Batam,

Lea Lindrawijaya Suroso, M.Pd
NIP. 19690124 199503 2 005

PREFACE

All praise and gratitude prayed to Allah SWT, because the blessings, mercy and grace The Author can resolve this report as one of the goals of the activities of Field Work Practice (PKL) which The Author do.

This report is based on data, knowledge and information that The Author get as long as do the Field Work Practice program at PT. Duta Computer. During The Author follow the learning in the industry, The Author gets a lot of knowledge that is very useful for Author. Author expects this report to be useful to the readers.

Therefore, on this occasion The Author is grateful profusely to my father, mother, and my beloved brothers who always provide both the moral and material support, as well as the industry that has given many contribute in the implementation of Industrial Work Practice, namely:

1. Mr. Rudy as Head Branch of PT. Duta Computer.
2. Mr. Ruslan as Head Chef Technician of PT. Duta Computer.
3. Dekcy Syhaputra, Alvino Simalango, Riced Novrianus as Mentor's Technician
4. All the Staff and Employees at PT. Duta Computer.
5. Mrs. Lea Lindrawijaya Suroso, M.Pd. as Headmaster of State Vocational High School 1 Batam.
6. Mr. Junaidi, S.Pd. as a coordinator of Industrial Work Practice program (PRAKERIN) Vocational High School 1 Batam.
7. Mrs. Suci Ramadhani, S.Pd. as Head of Computer and Network Department
8. Mr.Yarliansyah Agustian, S.ST. as my report advisor.
9. Mrs. Putri Wulandari, S.S. as Industry Guidance.
10. Mrs.Mike Musna, S.Pd. as Author teacher room XI TKJ Axioo.
11. And for all friends, that have educated and taught The Author during the learning was beginning until finished this report.

The Author realize this report still has many lacks. Because of that, The Author need the critique and the constructive suggestion for fullness of this report.

The Author also deeply apologize, if in this report readers find mistakes and words that unable to please. Hopefully this report could be useful for all of us.

Batam, 13th May 2019

Author

TABLE OF CONTENT

LIST OF PICTURES

CHAPTER I

INTRODUCTION

1.1 Background Selection of Titles

Version control systems are a category of software tools that help a software team manage changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

For almost all software projects, the source code is like the crown jewels - a precious asset whose value must be protected. For most software teams, the source code is a repository of the invaluable knowledge and understanding about the problem domain that the developers have collected and refined through careful effort. Version control protects source code from both catastrophe and the casual degradation of human error and unintended consequences.

Software developers working in teams are continually writing new source code and changing existing source code. The code for a project, app or software component is typically organized in a folder structure or "file tree". One developer on the team may be working on a new feature while another developer fixes an unrelated bug by changing code, each developer may make their changes in several parts of the file tree.

Version control helps teams solve these kinds of problems, tracking every individual change by each contributor and helping prevent concurrent work from conflicting. Changes made in one part of the software can be incompatible with those made by another developer working at the same time. This problem should be discovered and solved in an orderly manner without blocking the work of the rest of the team. Further, in all software development, any change can introduce new bugs on its own and new software can't be trusted until it's tested. So testing and development proceed together until a new version is ready.

Good version control software supports a developer's preferred workflow without imposing one particular way of working. Ideally it also works on any platform, rather than dictate what operating system or tool chain developers must use. Great version control systems facilitate a smooth and continuous flow of changes to the code rather than the frustrating and clumsy mechanism of file locking - giving the green light to one developer at the expense of blocking the progress of others.

Software teams that do not use any form of version control often run into problems like not knowing which changes that have been made are available to users or the creation of incompatible changes between two unrelated pieces of work that must then be painstakingly untangled and reworked. If you're a developer who has never used version control you may have added versions to your files, perhaps with suffixes like "final" or "latest" and then had to later deal with a new final version. Perhaps you've commented out code blocks because you want to disable certain functionality without deleting the code, fearing that there may be a use for it later. Version control is a way out of these problems.

Version control software is an essential part of the every-day of the modern software team's professional practices. Individual software developers who are accustomed to working with a capable version control system in their teams typically recognize the incredible value version control also gives them even on small solo projects. Once accustomed to the powerful benefits of version control systems, many developers wouldn't consider working without it even for non-software projects.

By far, the most widely used modern version control system in the world today is Git. Git is a mature, actively maintained open source project originally developed in 2005 by Linus Torvalds, the famous creator of the Linux operating system kernel. A staggering number of software projects rely on Git for version control, including commercial projects as well as open source. Developers who have worked with Git are well represented in the pool of available software development talent and it works well on a wide range of operating systems and IDEs (Integrated Development Environments).

1.2 The Purposes

The purpose of making this report are:

1. As readers can know what is Version Control System
2. As readers can know what are advantages of Version Control System.
3. As readers can know what is Git.
4. As readers can know what is GitHub and GitHub Desktop.
5. As readers can know what are advantages of using GitHub Desktop for project developing.

1.3 The Issues

In this part I will explain about the issues when making this report. There are five main problem:

1. What is Git and its advantages than another VCS?
2. What is GitHub?
3. What is GitHub Desktop?
4. Why GitHub more popular than other Git-repository hosting service?

CHAPTER II
COMPANY OVERVIEW

2.1 Background of The Company

PT. DUTA COMPUTER was founded in 1997 as a personal computer store by doing business directly. PT. The Duta computer is located at the Executive Center Complex Blok II No. 1-2 Jalan Laksamana Bintan, Panas River, Sungai Panas, Batam Kota, Kota Batam, Kepulauan Riau.PT. Duta computers now operate at 3 stores that house 34 employees, with 13 of them as technicians.

2.2 Vision and Mission

2.2.1 Vision

- improve the company's human resources so that it has a high level of professionalism.
- development or pioneer of quality information technology in the era of globalization

2.2.1 Mission

- Meet the needs of the community to achieve quality standards of the faculty by focusing on the integration of information technology in every aspect of human life.
- work in harmony as an overall business solution to meet the needs on the market today.
- Producing various types of computers or technological sophistication related to the needs of industry and society with highly competitive price and supply quality through professional management for customer satisfaction.

2.3 Field of Services

The company is involved in providing Hardware, Network and Security for business / office installations. types of services offered in the form of:

2.3.1 Hardware

Duta Computer provides its customers with the means to get various types of brands and brands of PCs, Servers and Accessories. For customers who have more specific needs, Duta Computer also allows customers to customize certain systems based on their needs and requirements. besides computer systems, Duta Computer also provides accessories such as printers, LCD monitors, modems, Sound & Video Cards, and others. As for inventory, Duta Computer provides ink and ribbons used in the most common printers available on the market. As a new addition, Duta Computer now holds the APC Brand distribution from UPSes, for more support for Batam customers who need power protection, this includes sales points and service centers, as well as drop-off generation points.

For the PC brand, Duta Computer also acts as a reseller for the Dell PC brand. This ensures that all requirements for Dell brand PCs and servers can be maintained directly from Batam, thereby cutting off more transportation, support time and costs from Jakarta. Like others, we also treat Dell's 3-year service guarantee for Batam and the surrounding area.

2.3.2 Maintenance Services and Support

Maintenance Services and support have always been a strong part of the company. Duta Computer is a retail store in Batam that only accepts one to one surrogate guarantee (with certain rules applicable) to its customers. For those items that are guaranteed one-to-one replacement cannot be applied, Duta Computer also provides a temporary replacement for goods, while awaiting a decision for the item to be repaired. Duta Computer has quite a lot of business running mission-critical production systems as customers, Duta Computer also has to apply certain rules to access this type of customer. Thus

is the reason for the 4 hour response to the standard method applied to the customer, where Duta Computer provides technical specialists within 4 hours of customer calls.

2.3.3 Network Design, Consulting and Security systems

From the last few years, Duta Computer also provides network and security systems. From small offices it is necessary to share printers for private companies that provide internet gateways to their customers, Duta Computer has provided everything. These services range from site visits, design, equipment procurement, and implementation, to testing, maintenance and warranty services. Especially for security systems, Duta Computer also holds distribution for Intotech, a Korean surveillance based on CCTV Camera System. This is a PC camera-based server, allowing the real time and size to depend only on the availability of HDD space.

With service as a focus we serve our customers, who rely on Ambassador Computer as a true partner in the success of their business. We not only provide them with the absolute best technology, products and services, but we also listen to them, and anticipate their future / business needs and get their business.

2.4 Organizational Structure

1 Rudy Sie as Managing Director and General Manager
2 Jerry as President Commissioner and Accounting and Finance
3 Zubirianto as Commissioner, IT Manager and Betwork Division
4 Ruslan as Human Resource Development
5 Doddy, Venny and Dewi as Marketing
6 Fusion and Dery as Purchasing
7 Yohanse and Password as a Store
8 Heru as IT Consultant
9 Susanti as Accounting
10 Lee Yong as Accounting and Receive

11 Surya as Finance
12 Theresia as Technical Supervisor
13 Lily as an Assistant Supervisor
14 Ruslan, Riced, Decy and Suryanto as Laptop and Computer Technicians
15 Leo and Alvino as Printer, Projector and Uninterruptible Power Supply (UPS) Technicians
16 Heru, Hendra PK and Indro as Server and Network Computer Technicians
17 Syaiful, Imron, Yunizar, Hendrik, Azman, Azmi, David and Suhelmi as Field Technicians
18 Zaharudin and hendra as Office Boy

2.5 Company Regulation

1. PT Ambassador Computer starts an activity starting at 08.00 to 17.00 and breaks from 12.00 to 13.00
2. Employees must enter work at the appointed hour.
3. Employees are required to wear neat, polite, and wear shoes.
4. Wealthman must comply with the specified hours of work.
5. Employees are prohibited from smoking during working hours or in the environment around the company.
6. Employees are prohibited from carrying out company property without the knowledge of the authorities.
7. Employees are strictly prohibited from damaging company property.

CHAPTER III
DISCUSSION

3.1 General Theory

3.1.1 Git

Picture 3.1 Git Logo

Git is a mature, actively maintained open source version control system project originally developed in 2005 by Linus Torvalds, the famous creator of the Linux operating system kernel. A staggering number of software projects rely on Git for version control, including commercial projects as well as open source. Developers who have worked with Git are well represented in the pool of available software development talent and it works well on a wide range of operating systems and IDEs (Integrated Development Environments).

3.1.2 GitHub

Picture 3.2 GitHub Logo

Though often used synonymously, Git and GitHub are two different things. Git is a particular implementation of version control originally

designed by Linus Torvalds as a way of managing the Linux source code. Other systems of version control exist though they are used less frequently. Git can be used to refer both to a particular approach taken to version control and the software underlying it.

GitHub is a company which hosts Git repositories and provides software for using Git. This includes 'GitHub Desktop' which will be covered in this report. GitHub is currently the most popular host of open source projects by number of projects and number of users.

Although GitHub's focus is primarily on source code, other projects, such as the Programming Historian, are increasingly making use of version control systems like GitHub to manage the work-flows of journal publishing, open textbooks and other humanities projects. Becoming familiar with GitHub will be useful not only for version controlling your own documents but will also make it easier to contribute and draw upon other projects which use GitHub.

GitHub Desktop will allow us to easily start using version control. GitHub Desktop offers a Graphical User Interface (GUI) to use Git. A GUI allows users to interact with a program using a visual interface rather than relying on text commands. Though there are some potential advantages to using the command line version of Git in the long run, using a GUI can reduce the learning curve of using version control and Git.

3.2 Advantages and Disadvantages of Using Git

3.2.1 Advantages

a. Performance

The raw performance characteristics of Git are very strong when compared to many alternatives. Committing new changes, branching, merging and comparing past versions are all optimized for performance. The algorithms implemented inside Git take advantage of

deep knowledge about common attributes of real source code file trees, how they are usually modified over time and what the access patterns are.

Unlike some version control software, Git is not fooled by the names of the files when determining what the storage and version history of the file tree should be, instead, Git focuses on the file content itself. After all, source code files are frequently renamed, split, and rearranged. The object format of Git's repository files uses a combination of delta encoding (storing content differences), compression and explicitly stores directory contents and version metadata objects.

b. Security

Git has been designed with the integrity of managed source code as a top priority. The content of the files as well as the true relationships between files and directories, versions, tags and commits, all of these objects in the Git repository are secured with a cryptographically secure hashing algorithm called SHA1. This protects the code and the change history against both accidental and malicious change and ensures that the history is fully traceable. With Git, you can be sure you have an authentic content history of your source code.

c. Flexibility

One of Git's key design objectives is flexibility. Git is flexible in several respects: in support for various kinds of nonlinear development workflows, in its efficiency in both small and large projects and in its compatibility with many existing systems and protocols.

3.2.2 Disadvantages

a. Git and GitHub Desktop could be difficult to learn
b. Some feature in GitHub.com need to pay for use like Wikis and more.

3.3 Working Procedure

3.3.1 Tool requirement for setup GitHub Desktop

Before we start, we need to bring the items and software as follows:

a) Item

1. Laptop / PC

Picture 3.3 Laptop / PC

b) Software

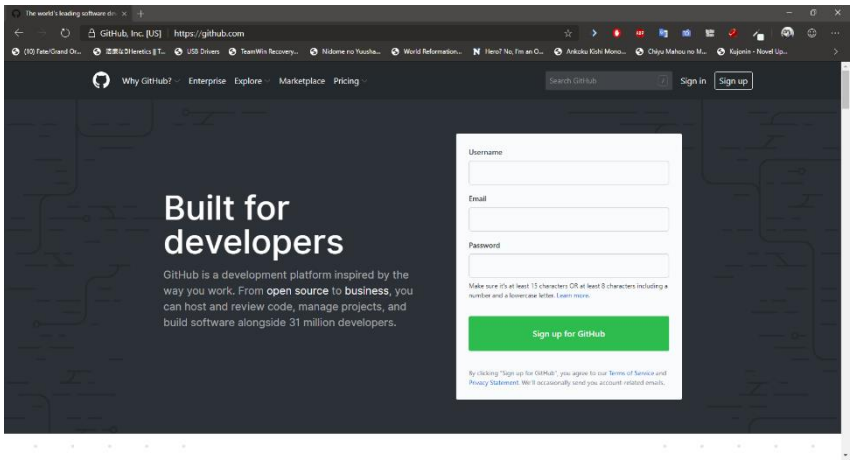
1. GitHub Desktop

Picture 3.4 GitHub Desktop

3.3.2 Make GitHub Account

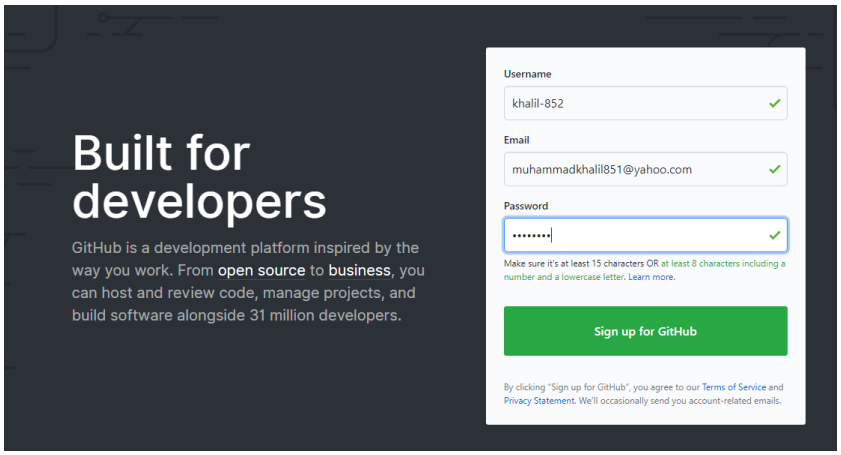
To use GitHub and make full use of GitHub Desktop, we first need to create a GitHub account:

1) Open https://github.com/ to create account



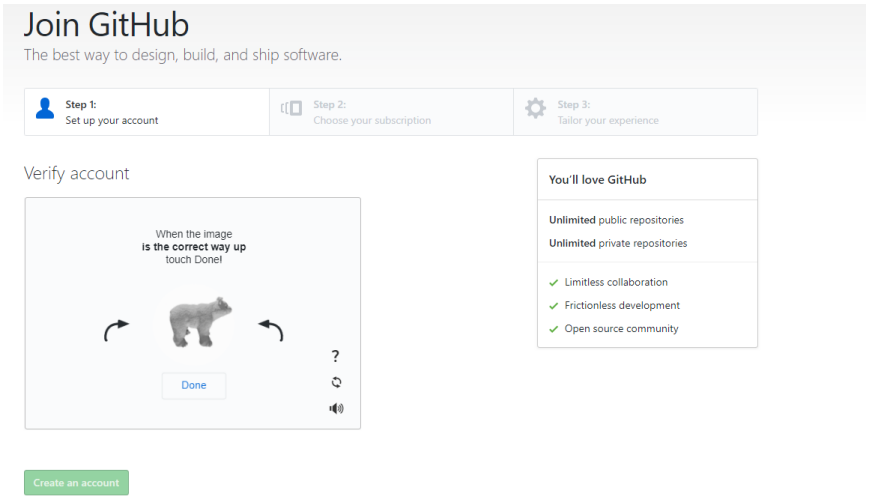
Picture 3.5 GitHub

2) Insert Username, email and password we want to use for account after that click sign up for GitHub



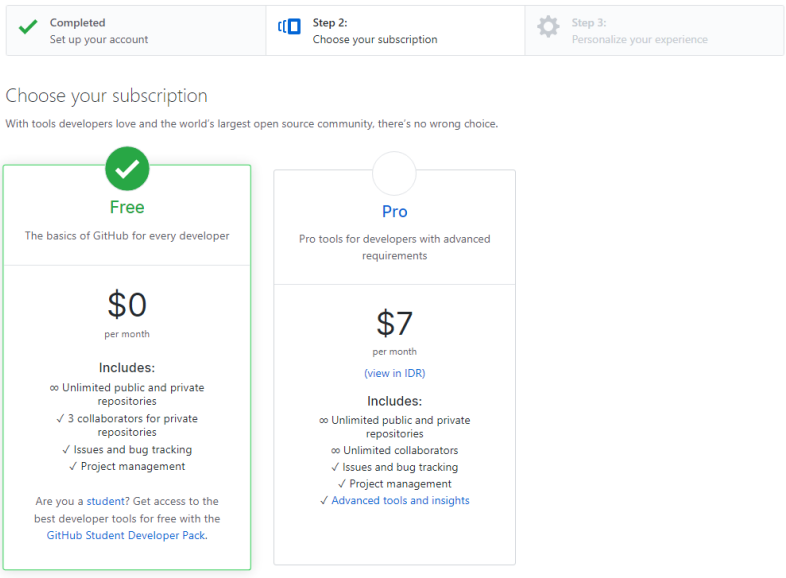
Picture 3.6 Sign Up

3) After that complete the step up for join GitHub. On step 1 we need verify account, they will give captcha for verify us is not a robot, solve the captcha then click done



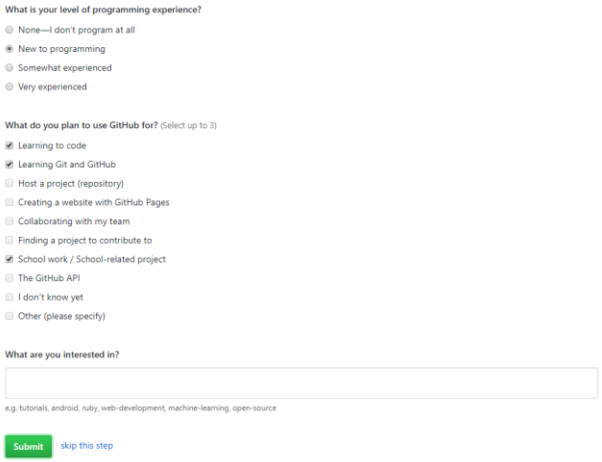
Picture 3.7 Step 1 Human validation

4) After that, choices subscription for account, on this report author use free subscription then click continue on the bottom page



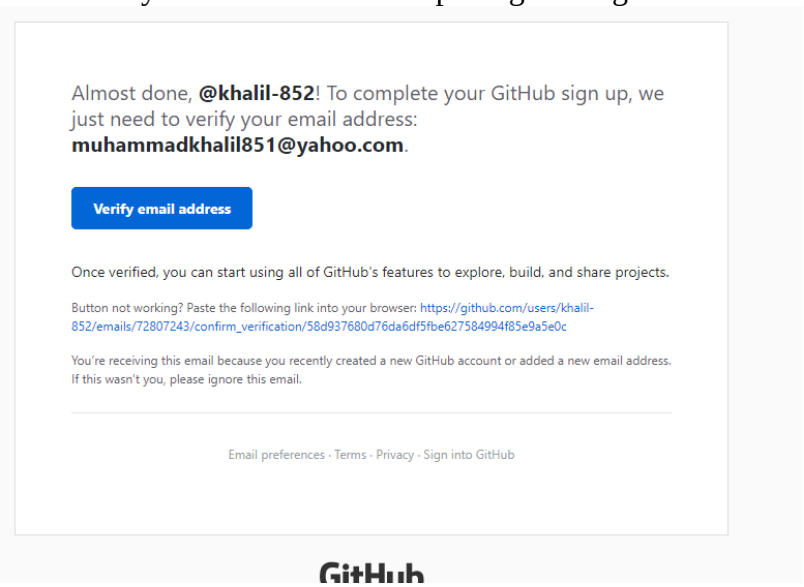
3.8 Step 2 Choose Subscription

5) On step 3, insert our experience on programming and what do you want to do for use GitHub or just skip it with the button Skip this step, after insert it click submit to continue

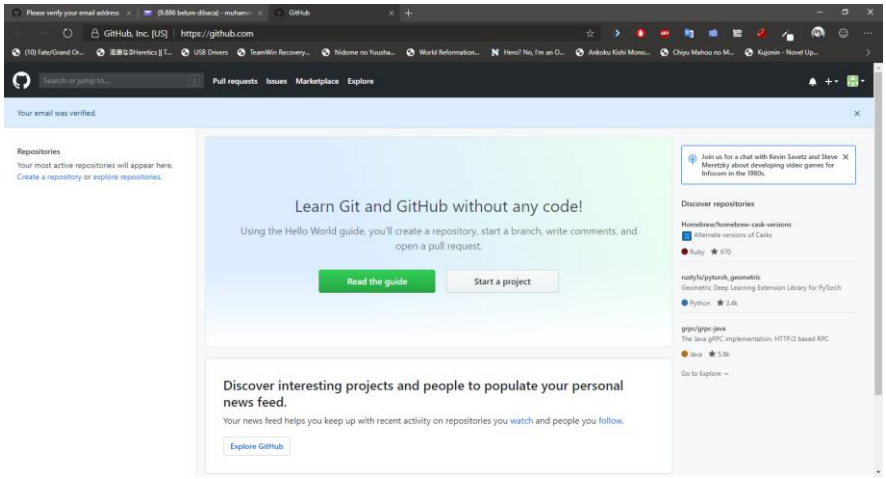


Picture 3.9 User Experience

6) After that GitHub will send an email to our email address, click verify email address to completing making our account



Picture 3.10 Email verify

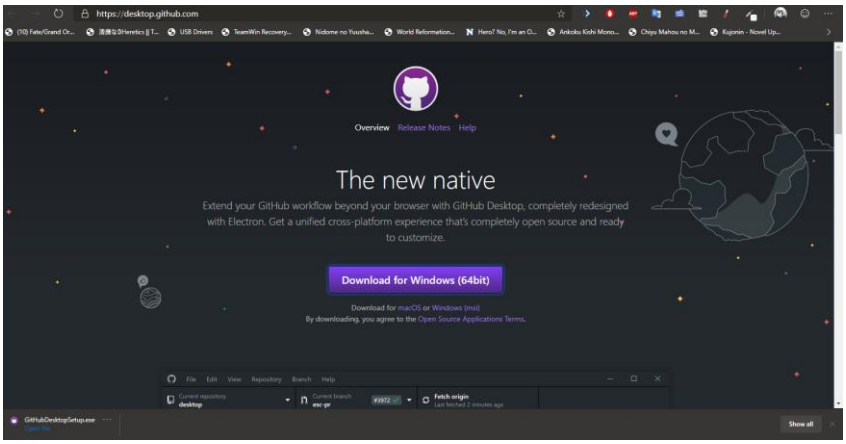


Picture 3.11 GitHub home page

3.3.3 Install GitHub Desktop

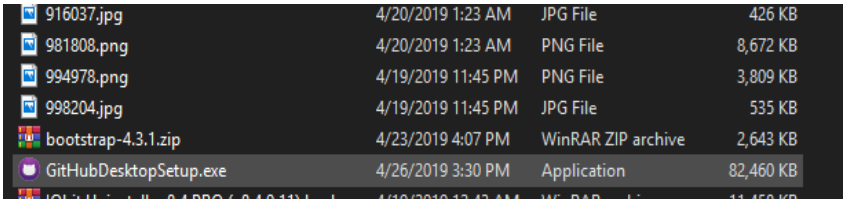
After create GitHub account, download and install GitHub Desktop:

1) Download GitHub Desktop from official Website:
   https://desktop.github.com/



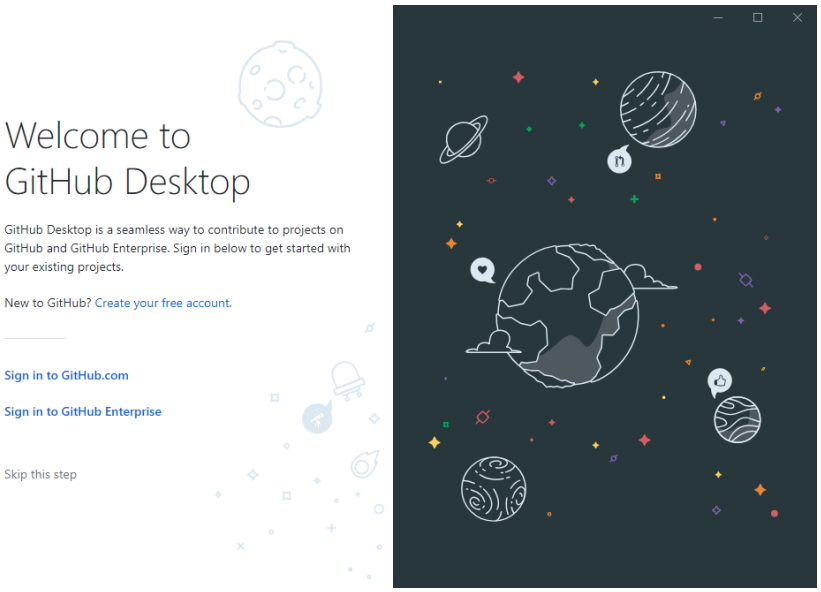
Picture 3.12 Download GitHub Desktop

2) Open the file to install GitHub Desktop, it will install automatically



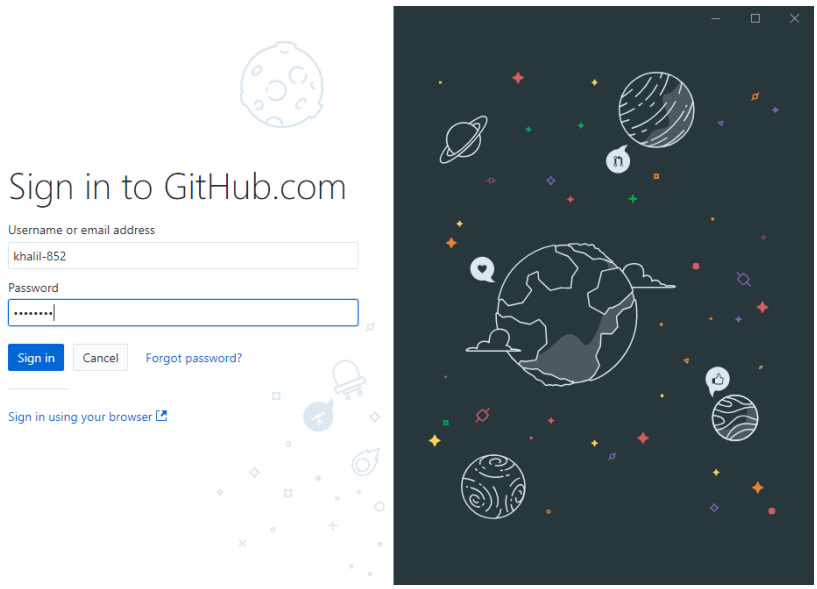
Picture 3.13 GitHub Desktop Installer

3) Click Sign in to GitHub.com to sign in with account we create

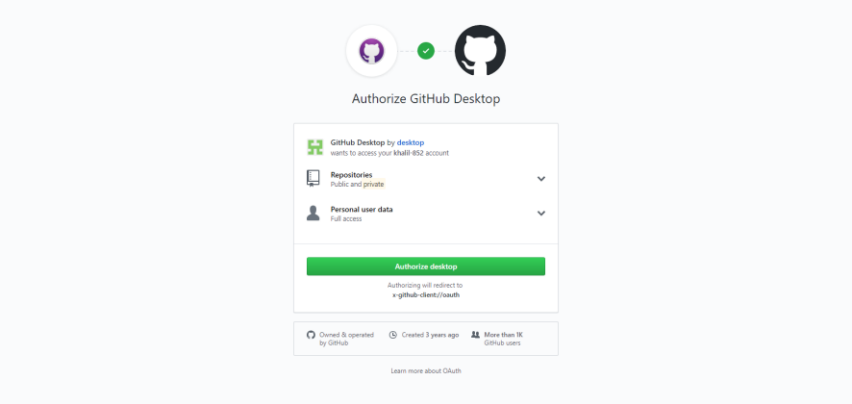


Picture 3.14 GitHub Desktop first launching

4) Insert Username or email address and our password then click sign in, we can use our browser instead it will open our browser then click authorize desktop to sign in with our browser

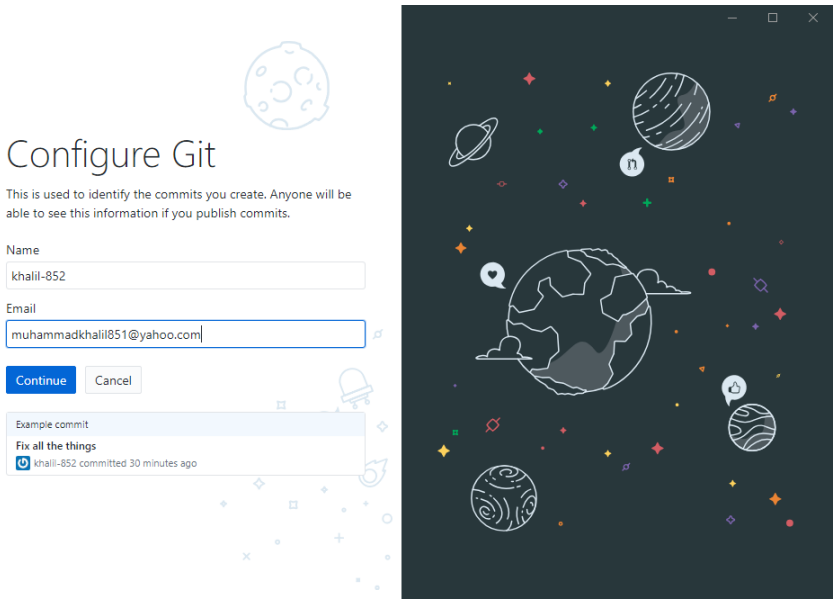


Picture 3.15 Sign in GitHub.com account



Picture 3.16 Authorize GitHub Desktop

5) Now configure Git just leave it as is if it automatically fills, if not, insert Name and email address for sync it with GitHub account (we can change it on setting), after that click continue to finishing installation GitHub Desktop.



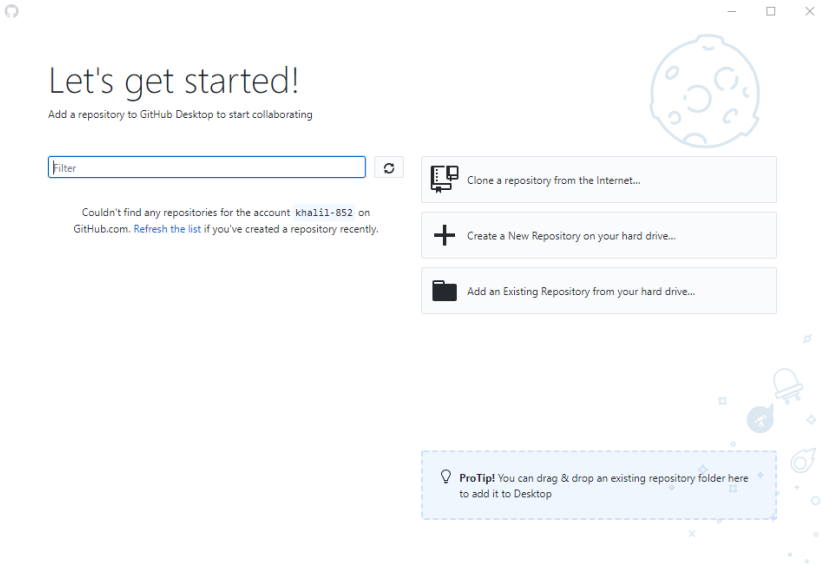
Picture 3.17 Configure Git for sync

3.3.4 Repository

Git repository is a virtual storage of your project. It allows you to save versions of your code, which you can access when needed.

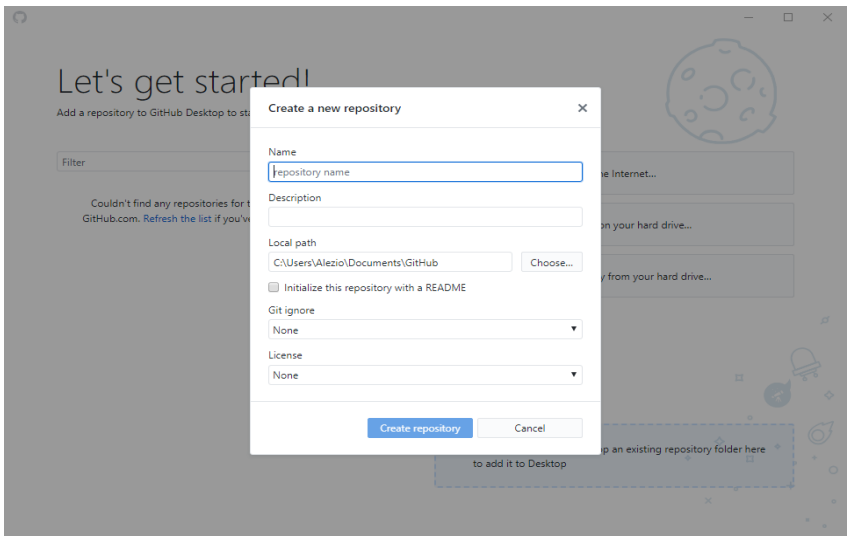
After install GitHub Desktop, make a new repository to GitHub desktop to start:

1.) Click “Create a New Repository on your hard drive” to make a new repository



Picture 3.18 Set-up Repository

2) There a few columns that need to fill for making a repository:



Picture 3.19 Create local Repository

- Name: Name of repository that will be make
- Description: describe the repository
- Local path: where will the repository will be making on local hard drive
- Git ignore: when the file is change or remove, git will ignore it and will not record it on versions
- License: this is need it when we want to make a copyright for our project

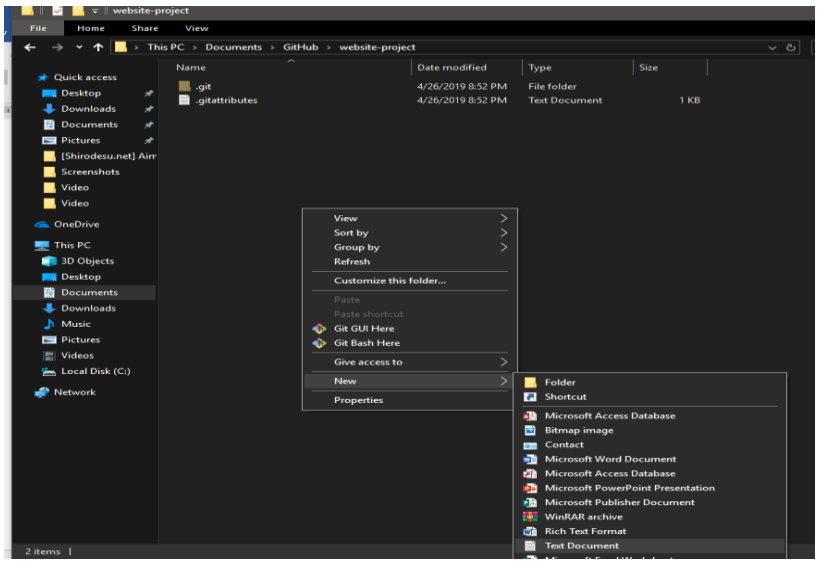
On this report, author will make a website repository for developing website, after filling out the column, click “Create Repository” to create repository.

3.3.5 Changes and Commit

GitHub Desktop have a slightly different ways of how command add, modified and staged work. In GitHub Desktop add, modified and staged is merged as one feature called Changes. Changes feature is that will automatically detect what files are added, changed and deleted in the local repository.

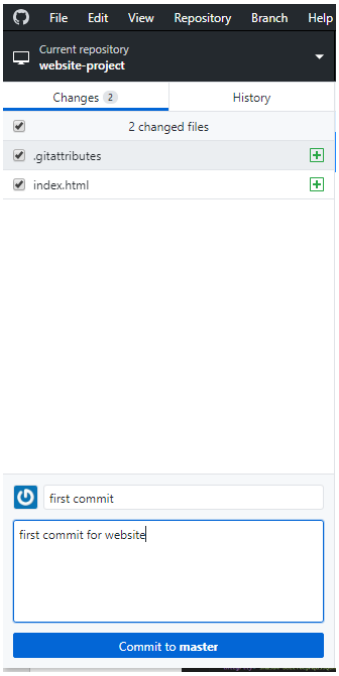
After create local repository we can add file to repository through directory we add-in to it.

1. Navigate to local repository we made, on this report author made local repository on “C:\Users\Alezio\Documents\GitHub\website-project”
2. After that make a file name index.html with right click then Next > Text Document, then rename it to index.html



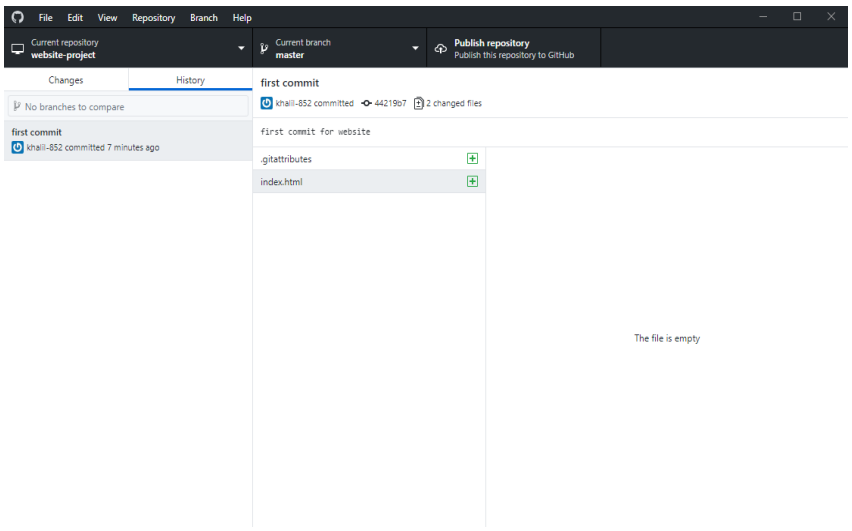
Picture 3.20 Create a new file

3. On GitHub Desktop, file that created or moving it to local repository will be added in Changes tab as 
4. To Commit what that change in repository, insert the title for commit that will be make in Summary that located in the bottom Changes tab, for the description it's optional, then click commit to master to commit.



Picture 3.21 "Changes" feature

5. To see what has been committed, click on History tab beside Changes tab



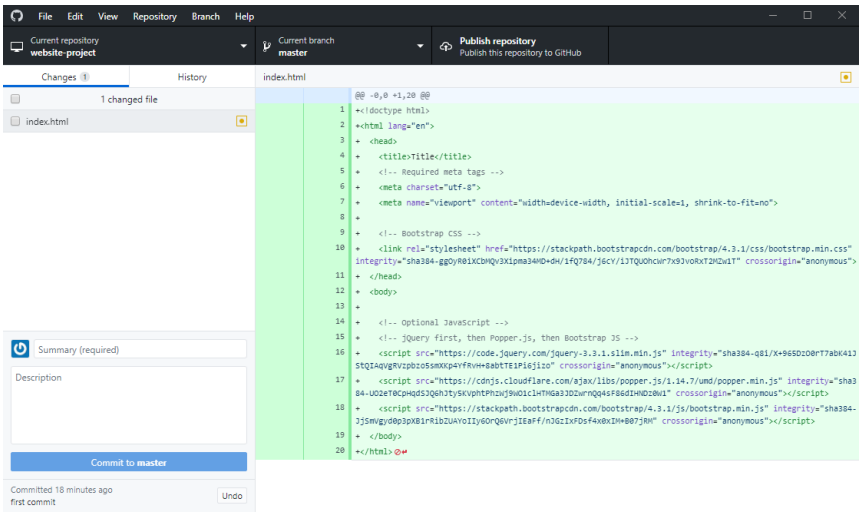
Picture 3.22 “History” feature

6. Now open index.html with text editor (Notepad, Visual studio Code Atom, etc.). On this report author use Visual Studio code as text editor.
7. Create basic HTML template on the empty file, the author uses Bootstrap as template on this report After making it save the file (ctrl + S)

```
<!doctype html>
<html lang="en">
  <head>
    <title>Title</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/
    iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
  </head>
  <body>
     <h1>Website Project</h1>
    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
    crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
    integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
    crossorigin="anonymous"></script>
  </body>
</html>
```

3.23 Coding line example

8. Now in the Changes tab there will be index.html file with  , this symbol is a sign that in the file there has been a change from the previous commit, on the right side you will see changes that have been occurred in the file:
   - Text with green highlight is a text that added to file
   - Text with red highlight is a text that have been remove from file



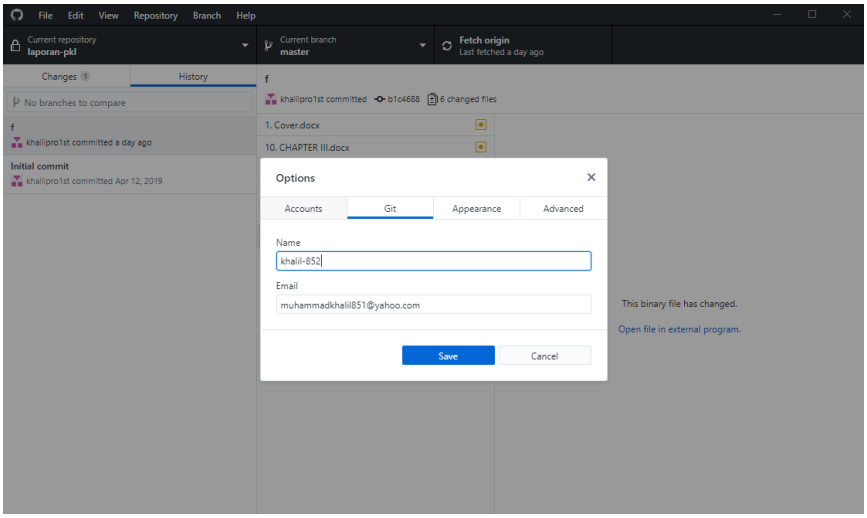
Picture 3.24 Line that have been changes

3.3.6 Remote Repository

Remote repositories are versions of your project that are hosted on the Internet or network somewhere. You can have several of them, each of which generally is either read-only or read/write for you. On this report author use GitHub as remote repository with GitHub account.

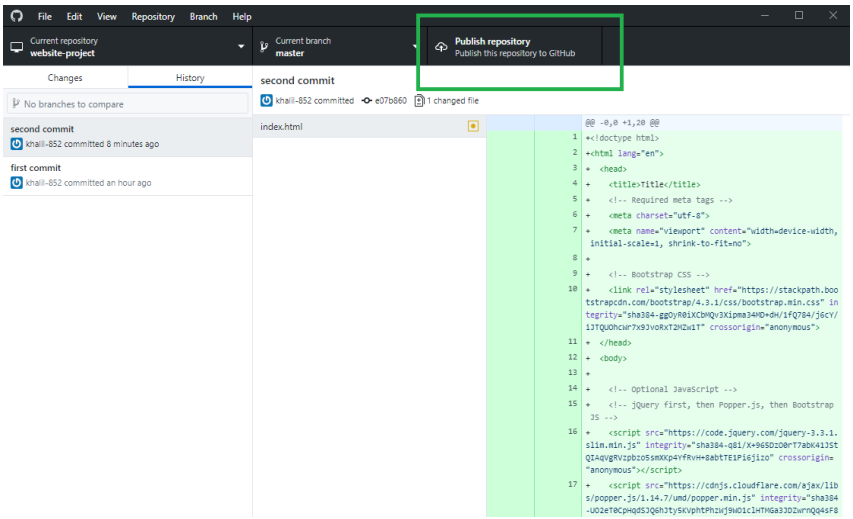
To add local repository to Remote repository on GitHub is with Push method. What the Push do is to push commits made on your local repository to a remote repository:

1) Open file > Options, Check if account on GitHub Desktop is same with that have been create earlier, then click Git to Check if Name and email is same with GitHub Account, after that click Save



Picture 3.25 GitHub Desktop Options

2) Now to push the Commit to GitHub, click Publish repository on top right GitHub Desktop or with Ctrl + P



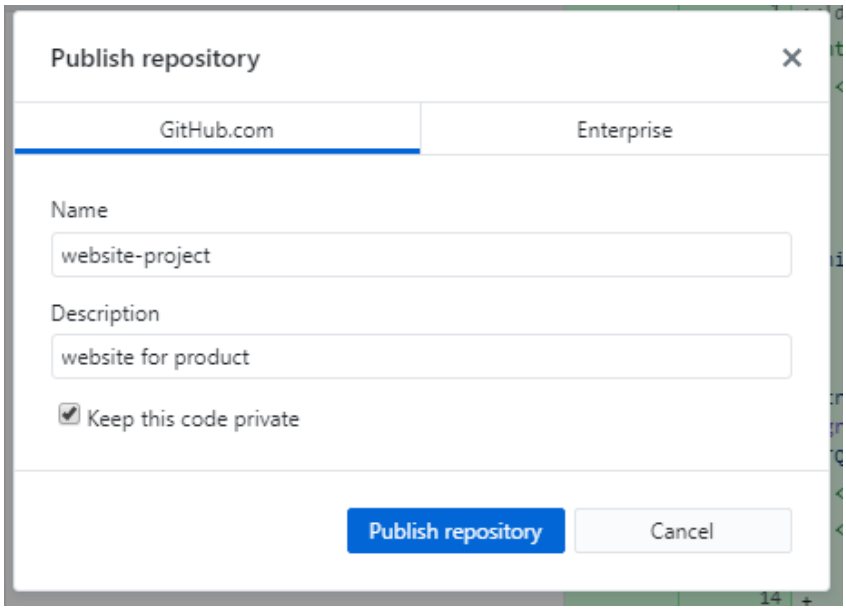
Picture 3.26 Publish repository Button

3) On Publish repository tab select GitHub.com, there will be column need to fill (it will automatically fill with information from local repository):

- Name: Name of remote repository on Github.com (try to make it same with local repository)
- Description: Description of repository (purpose, feature on it, etc.)

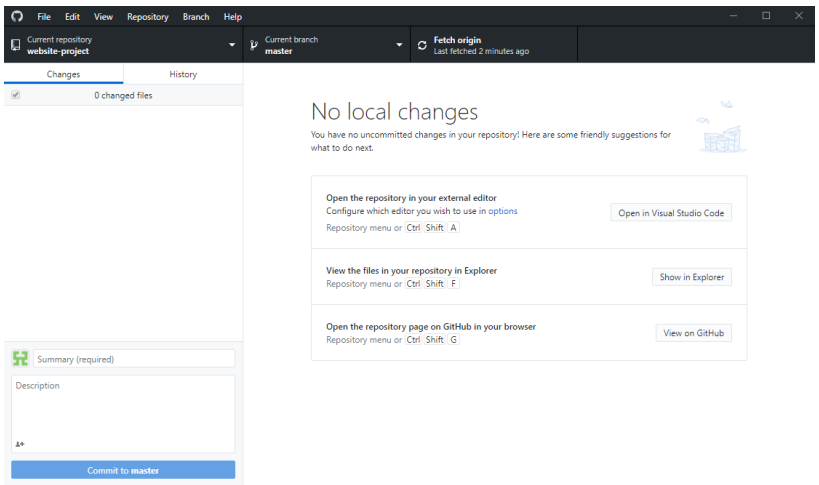
- Keep this code private: it will make repository private if it checked (only author or person with author permission can access it). On this report author unchecked it so anyone can access it

After that click Publish repository to push commit to GitHub.com



Picture 3.27 Create remote repository

4) If success Publish repository button will change to Fetch origin



Picture 3.28 Publish repository have changes to Fetch Origin

CHAPTER IV
CLOSING

4.1. Conclusion

In the last chapter of this report, the author has several conclusions that can be drawn from the author's report, namely:

1. Git is a mature, actively maintained open source version control system project originally developed in 2005 by Linus Torvalds. Git is the most reliable version control systems related to its performance, security and flexibility.
2. GitHub is a company which hosts Git repositories and provides software for using Git
3. GitHub Desktop is GitHub software that provides Graphical User Interface (GUI) to use Git. A GUI allows users to interact with a program using a visual interface rather than relying on text commands that's Git do.
4. GitHub is more popular because is a web hosting repository based on Git and have more feature like fork, hosting web for document their repository on their own domain (under Github.io), and more of it than other the web hosting repository. Plus, GitHub is base for some of popular open source project.

4.2. Suggestions

1. for bigger project, it's recommended to use GitHub Pro/GitHub Enterprise subscription for unlimited collaborators
2. for users that usually use command line like Linux distro, using GitHub Desktop GUI need time to get familiar with because some command is been merged with, like add, modified, remove command.

REFERENCE

SMKN 1 BATAM. May 2019. Journal Book. Batam, Riau Islands, Indonesia

Wikipedia, *GIT*. Accessed on March 15th 2019

https://en.wikipedia.org/wiki/Git

Wikipedia, *GitHub*. Accessed on March 16th 2019

https://en.wikipedia.org/wiki/GitHub

Atlassian, *what is version control*. Accessed on March 24th 2019

https://www.atlassian.com/git/tutorials/what-is-version-control

Atlassian, *what is git*. Accessed on March 23th 2019
https://www.atlassian.com/git/tutorials/what-is-git

Petanicode, *Tutorial belajar git untuk pemula*. Accessed on march 27th 2019

https://www.petanikode.com/tutorial/git/