# COMP1035 – Lab 01: Team Version Control

**Objectives:**
1. To collaboratively produce documentation in team git repository.
2. To demonstrate an understanding of the team-level control.
3. To successfully handle any merge conflicts that you may encounter.

---

## A. Getting Used to Git and Markdown – Using Command Line Interface (CLI)

### Git Task 1

**Step 1: Fork the repository (this task is for team leader ONLY)**

a. Log into https://csprojects.nottingham.edu.cn with your university ID and password (**ALL MEMBERS**).
b. Under "Projects", create a new project for the team by clicking on "New project" green button.
c. Write an awesome project name (which will be the team's name), make sure the "Visibility Level" is set to "Private".
d. Click on "Create project" green button. You can ignore all other options for now.
e. Next, check on the "**Lab01-HowToForkAndClone**" and follow "**Task A**" instructions to fork the repository template to your team's location.

****IMPORTANT**: Once the team leader complete the first step, the leader can now add your team member to your team repository. For CS internal server, search for team member names (**make sure the team member has log into the GitLab at least ONCE**) and add them with role "**Maintainer**". For public server, the team can search our email address and similarly, add us with role "**Maintainer**".

**Step 2: All members (on your individual computer): obtaining a local copy of the repo.**

a. Git clone **YOUR TEAM's** forked repository to wherever you want to work on your local machine (laptop). **DO NOT CLONE from the comp1035_fse/comp1035-template repository**.
b. Follow the "**Task B**" instructions in "**Lab01-HowToForkAndClone**".
c. Once the repository has cloned, type '`cd <team name folder>`' to navigate into the repository directory.
d. cd into the main folder, and you will see README.md.
e. Type '`git status`' to confirm that you have the latest version of the repository.

### Markdown Task 1

**Step 3: Create yourself a brief markdown profile in the "contributors" folder.**

a. Create a <`yourname`>.md (e.g., bryanlbg.md) in your "contributors" folder.

b. Add detail to this markdown file about you (see section **C1** in final page) – at least name, email address (as a link), and (ideally, but optionally) a photo. Feel free to add more detail about yourself as you wish such as your hobby, interests, achievements etc. A good sample from last year is provided in final page for your reference. You can edit the file in vim (**C2**) or nano (**C3**). Refer to "**Markdown Cheat Sheet**" for different format styles.
   i. You should put the photo in the images folder.

## Git Task 2

**Step 4: Synchronize a file with no conflict (hopefully).**

a. Git status – should only show your edited file.
b. Add your changes to the staging area by typing '`git add <your-profile>.md`'.
c. Commit your changes by typing '`git commit -m "<message about adding your profile>"`'.
   i. You should give your commit a sensible message like "Documentation: Updated Bryan's profile in the "contributors" folder". You can read the "**Writing Good Commit Messages**" for reference under Expected Readings in Week 01.
d. Push your changes to the remote repo by typing '`git push`'.
e. You are all editing different files, this should all happen fine.
   i. If not, then you might have to git pull first, and merge. Type "`git pull`" to bring the latest changes from the server (that other people have pushed) down to your repository. So, you can make sure it is merged first locally before you push yours up to the server.
      1. Refer to lecture or any online material you like as a guide.

## Markdown Task 2

**Step 5: Add a link to your markdown profile in the main README.md.**

a. In the README.md starter file, is a space for a list of contributors. Here you should add your name as a link in the list, that links to your profile (the link should be referencing link "`contributors/<yourname>.md`" – rather than a full https:// style link).
   i. This is going to create a situation where 6 or 7 of you edit the same file, and it might create a conflict that needs merging – see next step.

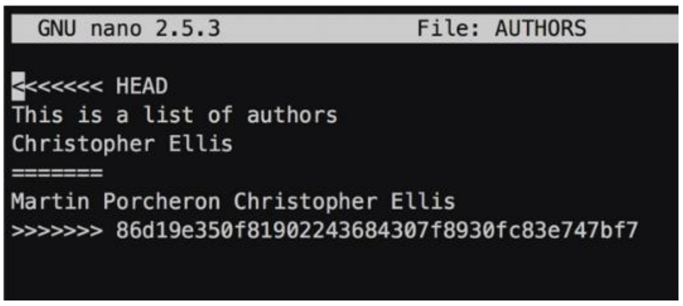
## Git Task 3

**Step 6: Synchronize a file with conflicts.**

a. Repeat Git Task 2 but for the main project README.md.
b. If you get any merge conflict warnings, fetch down the latest version of the report using the command '`git pull`' – it will attempt to merge it. If this gives further merge error messages use '`git status`' to view any conflicts.
   i. You need to edit the file and decide which bits you want to keep.
c. Then try pushing your changes to the remote repository again.

**Hint**: Handling merge conflict is hard – you may want to watch the demonstration video on this task at Lecture 01 or referring to some online resources.

If using nano, a merge conflict will look something like this:



```
  GNU nano 2.5.3                 File: AUTHORS

<<<<<<< HEAD
This is a list of authors
Christopher Ellis
=======
Martin Porcheron Christopher Ellis
>>>>>>> 86d19e350f81902243684307f8930fc83e747bf7
```

This lines between <<<<<<< HEAD and ======= represent the local changes you have done whilst the ======= to >>>>>>> represent what has been introduced in the same area by someone else.

You need to delete the <<<<<<<, ======= and >>>>>>> lines and merge the remaining lines to be what you want. Add then exit nano/vim for it to merge/push.

### Finishing Activity

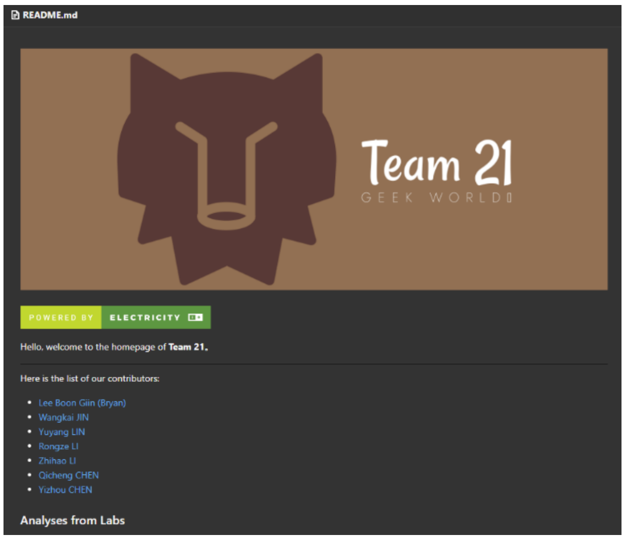
When everyone is done with the steps above, they should git pull, so that you all have the finished version!

---

## B. Team Activity – Setup Repository for Your Team

You should do this as a team.

1. You can edit the main README.md page as much as you want to be about your team. You can see an example below from last year.
2. Create milestones, labels, and default boards for your team.
3. Try creating, labelling, and closing issues.
4. You will be using the main README.md to link to work that you have done for the whole semester – both work from labs, and links to your coursework submissions. Consider creating headings for these things.

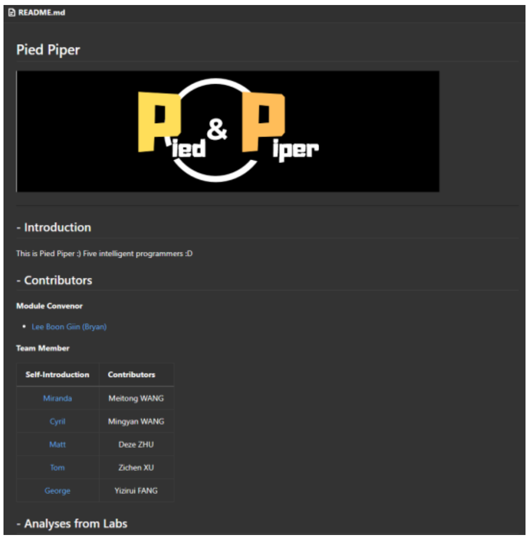

README.md

Hello, welcome to the homepage of **Team 21.**

Here is the list of our contributors:

- Lee Boon Giin (Bryan)
- Wangkai JIN
- Yuyang LIN
- Rongze LI
- Zhihao LI
- Qicheng CHEN
- Yizhou CHEN

Analyses from Labs


README.md

Pied Piper

- Introduction

This is Pied Piper :) Five intelligent programmers :D

- Contributors

**Module Convenor**

- Lee Boon Giin (Bryan)

**Team Member**

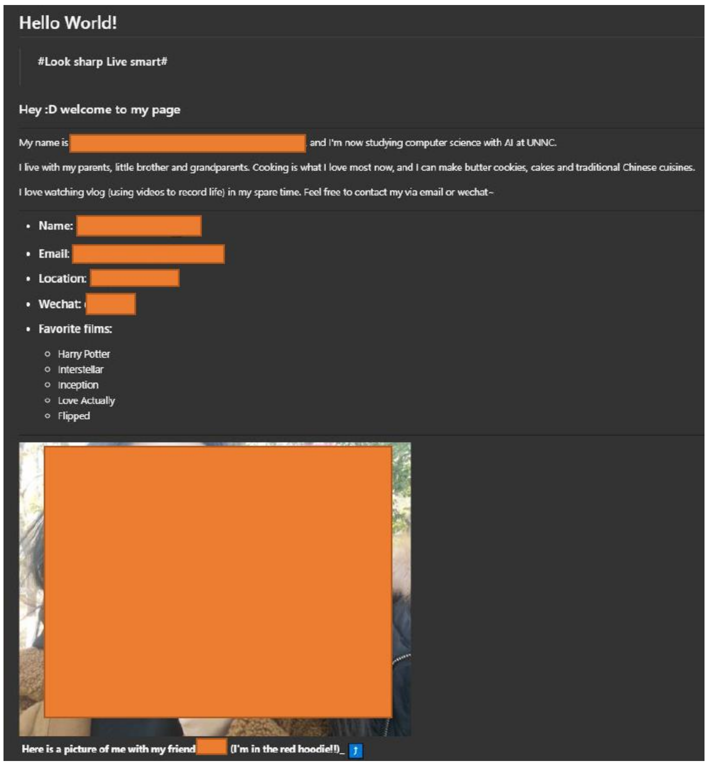
| Self-Introduction | Contributors |
|---|---|
| Miranda | Meitong WANG |
| Cyril | Mingyan WANG |
| Matt | Deze ZHU |
| Tom | Zichen XU |
| George | Yizirui FANG |

- Analyses from Labs

---

## C. Guides List

G1. GitLab Markdown Documentation
G2. Vim for Beginner Tutorial
G3. The Beginner's Guide to Nano Editor
G4. Git Installation
G5. Setting SSH Key in GitLab


Hello World!

> #Look sharp Live smart#

**Hey :D welcome to my page**

My name is [redacted] and I'm now studying computer science with AI at UNNC.

I live with my parents, little brother and grandparents. Cooking is what I love most now, and I can make butter cookies, cakes and traditional Chinese cuisines.

I love watching vlog (using videos to record life) in my spare time. Feel free to contact my via email or wechat~

- **Name:** [redacted]
- **Email:** [redacted]
- **Location:** [redacted]
- **Wechat:** [redacted]
- **Favorite films:**
  - Harry Potter
  - Interstellar
  - Inception
  - Love Actually
  - Flipped

[redacted]

**Here is a picture of me with my friend** [redacted] **(I'm in the red hoodie!!)_**

Good sample of contributor profile from last year.

---