CS673 Software Engineering

Lab 1 - Set Up Git

## Description:

The purpose of this lab is to make sure that all students get their Git/GitHub set up for their projects and can perform basic Git operations.

## Instructions:

1. Set up a GitHub account if you don’t have one already. Go to github.com and follow the instructions to set up your account.
2. Accept the project-assignment invitation. If you are the team leader or configuration leader who is in charge of your team’s GitHub repository, you can create a new team for your team, and it will create a repository for your team. Otherwise, you can simply join your team.
3. Install and set up Git on your local machine. Git is usually preinstalled on Mac and Linux. But if you are using Windows or want a GUI client, you will need to install it separately. Git comes with built-in GUI tools at <https://git-scm.com/downloads>. There are also some third-party tools. If you are not familiar with the command-line interface, a popular option is to use **GitHub Desktop**, since we will use GitHub for our projects (<https://desktop.github.com/>).
4. If you are not familiar with Git, please go through some Git tutorials. There are many online tutorials, such as <https://git-scm.com/docs/gittutorial> (this one is from the official Git site), <https://try.github.io/> (this one is from GitHub), and <https://github.github.com/training-kit/downloads/github-git-cheat-sheet.pdf> (a cheat sheet of Git commands).

Take some screenshots of the following steps and include them into your lab report:

1. Clone your team’s GitHub repository into your local machine. You can use either the Git command (git clone) or any Git GUI app such as Github Desktop.
2. If you are the team leader or configuration leader who is in charge of your team’s GitHub repository, create a new branch lab1. Other students can simply check out the lab1 branch using either the Git command (git checkout lab1) or any Git GUI app.
3. Create/Change the *team.md* file in the lab1 branch to add a paragraph about yourself, and commit the change to your local repository using either the Git command (git commit) or any Git GUI app. The *team.md* file is a simple markdown file. You can find more information about basic markdown syntax here: <https://www.markdownguide.org/basic-syntax/>.
4. Push your commit to your team’s remote GitHub repository using either the Git command (git push) or any Git GUI app. If there is any conflict, you will need to manually merge and solve it.
5. Go to the remote GitHub repository webpage and check whether your change was successful.
6. After everyone in your group has updated the file *team.md* and committed the change to the branch lab1, and the project leader has reviewed the file, if you are in charge of your team’s GitHub repository, you shall merge the lab1 branch into the main branch and then delete the lab1 branch.
7. Submit a lab report on Blackboard that includes the following sections:
   1. Your GitHub project URL and username
   2. Screenshots to show how you use Git to modify the file in the repository
   3. A summary paragraph about your experience with Git/GitHub and any issues you encountered
   4. A list of **at least three other commands** or functionalities from Git/GitHub that you may use later, with a brief explanation of each.