Your Team’s Version Control Work Flow

|  |  |
| --- | --- |
| Names:  Jacob Smith, Harmanbir Dhillon, Jonathan Chiu, Yu Ting Hsu, Alexander Sui | Team #  24 |

## Learning Objectives:

* Review basics of version control using git <https://git-scm.com/book/en/v2>
* Review repo hosting, eg. github <https://guides.github.com/activities/hello-world/>
* Review GIT, familiarize with common ways to use GIT in collaborative way

<https://www.atlassian.com/git/tutorials/comparing-workflows#gitflow-workflow>

* Learn the "GitFlow" work flow:

<http://nvie.com/posts/a-successful-git-branching-model/>

## Steps

1. Rename this file, and fill in boxes.
2. Review git, and discuss pros and cons of the various workflows with your team.
3. Agree on workflow (we suggest "GitFlow" or “Feature-Branch”). Write it out in your own words. State the steps clearly. Talk through it in your team. Make sure all team members have same understanding.

Questions to ask yourselves:

* How will everyone branch?
* Will you do local merges? Or pull-requests?
* Who will approve pull-requests?
* How will “merged” (or approved) pull-requests be communicated to the team?

Our Team's Agreed Workflow:

|  |
| --- |
| From here on out, we will review our git workflow to have the master/origin branch with a developer branch coming off it. We will all push to the develop branch and one of us will push to master when we decide there are permanent solutions implemented. Only the team leader (Jacob) will approve and communicate all pull-requests. |

1. As a team, go through the motions of creating, and developing, a new project, using your agreed workflow. Every team member, in a local directory, must create some file (like "bob.txt"), and contribute it to the repository.

Example: If you are using Gitflow, it might look like this:

* Someone create repo on github with “master”, and “develop”.
* Each contributor clone from GitHub
* Each contributor should branch from "develop"
* Do some work (commit)
* Each contributor's feature branch should merge (or pull-request) into "develop"
* Repeat last 2 steps a few times (optional)
* Finally, the "develop" branch should merge (or pull-request) into "master"

1. Show a Network Graph, which **demonstrates your workflow** clearly. In GitHub, this would be under the “Insights” tab, and “Network” sidebar menu.

Our Network Graph

|  |
| --- |
| Paste image of graph here:  Macintosh HD:Users:JacobSmith:Desktop:Screen Shot 2018-05-02 at 9.45.33 AM.png |

## Submission

Submit this document to D2L.