GIT SESSION

MOZILLA CAMPUS CLUB, NSIT

Introduction to GIT

Version Control System is a system that records changes to a file or a set of files over time so that it can be recall specific versions later.

Git is a distributed version control system for tracking changes in a project and co-ordinating work among multiple people primarily used for source code management in software development

The Git stores the information about the project in a data structure called a repository. The basic Git workflow goes something like this:

1. You modify files in your working tree.

2. You selectively stage just those changes you want to be part of your next commit, which adds only

those changes to the staging area.

3. You do a commit, which takes the files as they are in the staging area and stores that snapshot

permanently to your Git directory.

Installing git on your system:

Ubuntu:

sudo apt-get install git

Windows:

Download and setup from <http://git-scm.com/download/win>

Mac:

brew install git

For Setting up for the first time you need to tell the git system who you are through the commands

* git config --global user.name "name"
* git config --global user.email "email@example.com"

You typically obtain a Git repository in one of two ways:

1. You can take a local directory that is currently not under version control, and turn it into a Git repository

* git init

2. You can clone an existing Git repository from elsewhere

* git clone <url>

To stage a file is to prepare it for a commit. Because git exposes this action to the users control it allows you to create partial commits, or to modify a file, stage it, modify it again, and only commit or revert to the original modification. Staging allows you finer control over exactly how you want to approach version control.

For checking the staging area

* git status

To add/remove a file to staging area

* git add <filename> / git rm <filename>

To check what things have been differed since last commit

* git diff

To commit to Git

* git commit -m "message"
* git commit

opens text editor in terminal

add message to line without # symbol after pressing i on vim or just by simply clicking in nano

to exit vim press ESC then press: x

To see the history of repository

* git log

After doing all the necessary changes in a project it's finally ready to be uploaded to internet for the world to see

To push the Git repository to GitHub profile

* git push <remote> <branch>

and then enter the id/password for you GitHub profile for authentication. (might ask for it only once especially on windows)

A branch in Git is simply a lightweight movable pointer to one of these commits. The default branch name in Git is "master". As you initially make commits, you're given a "master" branch that points to the last commit you made.

For adding a new branch

* git checkout -b "branch\_name”

To check the list of branches

* git checkout –a, or
* git branch

If some changes are made on the GitHub by some other people and your local repository is a atleast one commit behind the origin then the newest version of the git repository can be pulled

* git pull <remote>