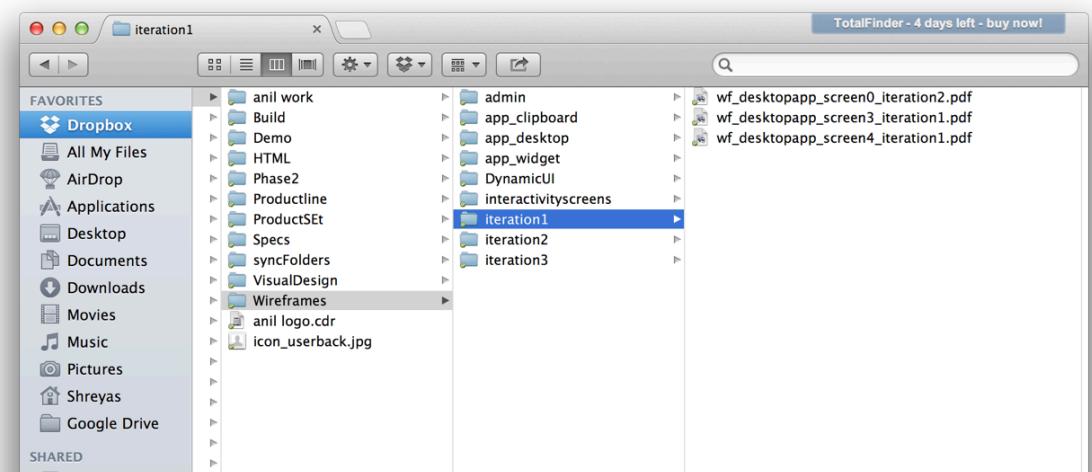


CSC 443: Web Programming

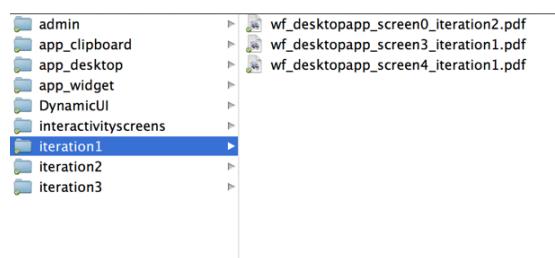


Workflow
an opinionated discussion

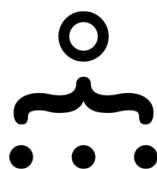
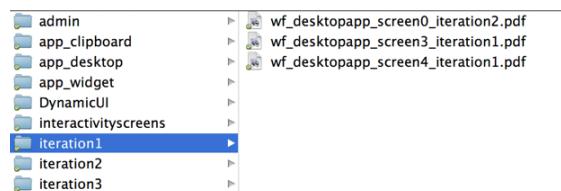
We've all been here ...



We've all been here ...

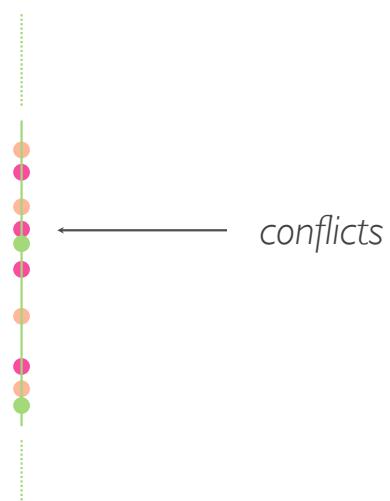


We've all been here ...

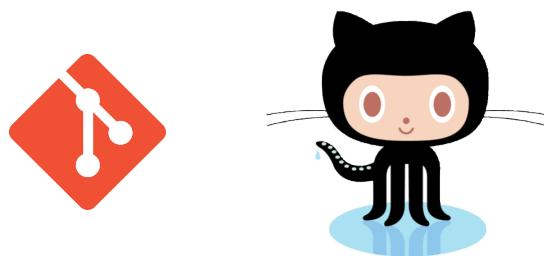


Problems

- multiple versions
- many contributors
- ...



Attempted Solutions



git + Github

a better workflow



git

git is an open source, distributed
version control system designed
for speed and efficiency



git

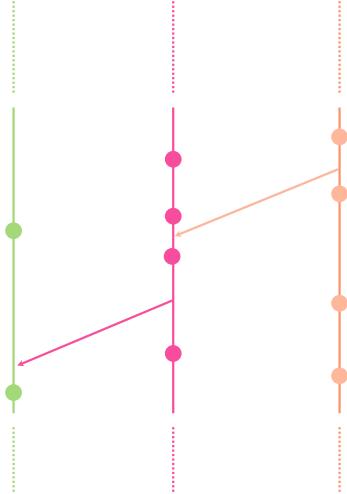
git is an open source, distributed
version control system designed for
speed and efficiency





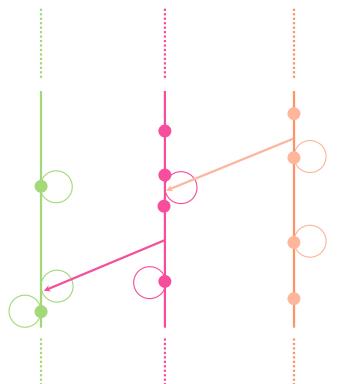
git

git is an open source, ***distributed*** version control system designed for speed and efficiency



git

git is an open source, distributed version control system designed for ***speed and efficiency***

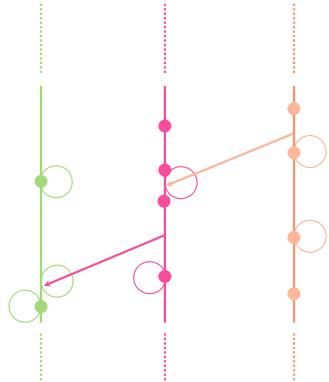


(almost) everything is local



No Network needed for

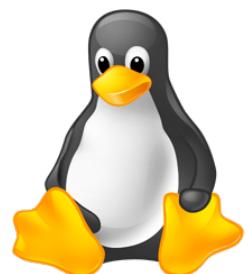
- Performing a diff
- Viewing file history
- Committing changes
- Merging branches
- Obtaining any other revision of a file
- Switching branches



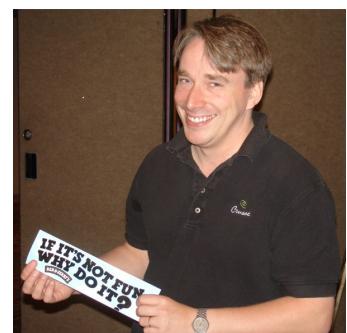
(almost) everything is local



git



git is an **open source**, distributed
version control system designed
for speed and efficiency



More Advantages

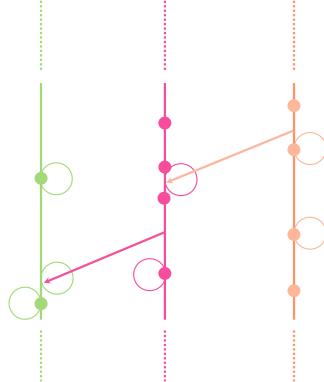
<http://bit.ly/git-advantages>

<http://bit.ly/git-linus-intro>



.. and

- No .svn directories
- Independence



installing git

<http://git-scm.com/downloads>

The entire [Pro Git book](#) written by Scott Chacon is available to read online for free. Dead tree versions are available on [Amazon.com](#).

Downloads

Latest stable release
1.8.1.1
Release Notes (2013-01-14)

Download for Mac

Mac OS X Windows

Linux Solaris

Older releases are available and the [Git source repository](#) is on GitHub.

GUI Clients

Git comes with built-in GUI tools ([git-gui](#), [gitk](#)), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

Git via Git



getting setup ...

```
$ git config --global user.name "<YOUR_NAME>"
```

```
$ git config --global user.email "<YOUR_EMAIL>"
```

or edit them in `~/.gitconfig`



lets dive in..

- Creating New Repo
- Committing
- Branching/Merging
- Inspection
Cloning a
Repo
Updating

The screenshot shows a web browser window with the URL gitref.org/about.html. The page title is "Git Reference". The navigation bar includes links for "Reference", "About", and "Site Source". The main content area is titled "Git Reference" and contains several sections with lists of Git commands:

- Getting and Creating Projects**: init, clone
- Basic Snapshotting**: add, status, diff, commit, reset, rm, mv, stash
- Branching and Merging**: branch, checkout, merge, log, tag
- Sharing and Updating Projects**: fetch, pull, push, remote
- Inspection and Comparison**: log, diff

A sidebar on the right says "WHO DID THIS? The Git Reference site was put together by the GitHub team."



lets dive in..

<http://git-scm.com/docs>

The screenshot shows the official Git documentation website. The top navigation bar includes links for Stock photos, Favs, Prototyping, Web, Usability, Useful Sites, OS, Platforms, Projects, Awesome, and Berkeley. The main content area features the Git logo and the tagline "fast-version-control". On the left, there's a sidebar with links for About, Documentation (Reference, Book, Videos, External Links), Downloads, and Community. The right side is titled "Reference" and contains a tree view of Git commands:

- Setup and Config
 - config
 - help
- Getting and Creating Projects
 - init
 - clone
- Basic Snapshotting
 - add
 - status
 - diff
 - commit
 - reset
 - rm
 - mv
- Branching and Merging
 - branch
 - checkout
- Debugging
 - bisect
 - blame
 - grep
- Email
 - am
 - apply
 - format-patch
 - send-email
 - request-pull
- External Systems
 - svn
 - fast-import
- Administration
 - clean
 - gc
 - fsck

- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating



Git Commands

- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

\$ git init

\$ mkdir workspace

\$ cd workspace

\$ ls -al

\$ git init

\$ ls -al

Initialized empty Git repository in ./workspace/.git/

drwxr-xr-x 10 Shreyas staff 340 Jan 15 17:22 .git



Git Commands

- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

```
$ cd .git
```

```
$ tree -a
```

```
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace/.git
$ tree -a
.
├── HEAD
├── branches
├── config
├── description
└── hooks
    ├── applypatch-msg.sample
    ├── commit-msg.sample
    ├── post-update.sample
    ├── pre-applypatch.sample
    ├── pre-commit.sample
    ├── pre-rebase.sample
    └── prepare-commit-msg.sample
        └── update.sample
    └── info
        └── exclude
    └── objects
        ├── info
        └── pack
    └── refs
        ├── heads
        └── tags
9 directories, 12 files
```



Git Commands

- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

```
$ cd ..
$ touch testfile.md
```

\$ git status

branch name →

staging area →

```
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace
$ git status
# On branch master
#
# Initial commit
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       testfile.md
nothing added to commit but untracked files present (use "git add" to track)
```



Git Commands

- Creating New Repo
- **Committing**
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

\$ git add <filename>

\$ git status

staging area →

```
$ git add testfile.md
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace
$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   testfile.md
#
```



Git Commands

- Creating New Repo
- **Committing**
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

\$ git commit

\$ git status

your commit msg →

```
0 Sachin_guide_me
1 # Please enter the commit message for your changes. Lines starting-
2 # with '#' will be ignored, and an empty message aborts the commit.-
3 # On branch master
4 #
5 # Initial commit
6 #
7 # Changes to be committed:-
8 #   (use "git rm --cached <file>..." to unstage)
9 #
10 #       new file:   testfile.md
11 #
^
^
^
^
```



Git Commands

- Creating New Repo
- **Committing**
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

\$ git reset

(undo add)

changes staged →

changes unstaged →

```
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*
$ git add testfile.md
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*
$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   testfile.md
#
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*
$ git reset
Unstaged changes after reset:
M      testfile.md
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*
$ git status
# On branch master
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   testfile.md
#
no changes added to commit (use "git add" and/or "git commit -a")
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*
```



Git Commands

- Creating New Repo
- **Committing**
- Branching/Merging
- Updating
- Inspection
- Cloning A Repo

\$ git reset --soft HEAD^

(undo a commit and redo)

\$ git reset --hard HEAD~3

Rewind the branch to get rid of last three commits

learn more here :

<http://git-scm.com/docs/git-reset>

also refer

git revert : <http://git-scm.com/docs/git-revert.html>

git stash : <http://git-scm.com/docs/git-stash>



Git Commands

- Creating New Repo
- **Committing**
- Branching/Merging
- Inspection
- Cloning a Repo
- Updating

`$ git rm <filename>`

*remove a file from the repo
as expected, you will need to commit the change*

`$ git mv <filename>`

*move/rename a file from the repo
as expected, you will need to commit the change*



Git Commands

- Creating New Repo
- Committing
- **Branching/Merging**
- Inspection
- Cloning a Repo
- Updating

`$ git branch`

list of branches

`$ git branch <branchname>`

new branch

`$ git checkout <branchname>`

switch



Git Commands

- Creating New Repo
- Committing
- **Branching/Merging**
- Inspection
- Cloning a Repo
- Updating

```
branch list →      Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*  
$ git branch  
* master  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*  
$ git branch dev  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*  
$ git branch  
  dev  
* master  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master*  
$ git checkout dev  
M  testfile.md  
Switched to branch 'dev'  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on dev*  
$ git branch  
* dev  
  master  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on dev*
```

new branch dev →

switch branch →



Git Commands

- Creating New Repo
- Committing
- **Branching/Merging**
- Inspection
- Cloning a Repo
- Updating

\$ git merge <branchname>

merge branches

branch merge →

```
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on dev  
$ ls  
anotherfile.md testfile.md  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on dev  
$ git branch  
* dev  
  master  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on dev  
$ git checkout master  
Switched to branch 'master'  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master  
$ ls  
testfile.md  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master  
$ git merge dev  
Updating e63df69..d0f3f3f  
Fast-forward  
  testfile.md | 3 +-  
  1 file changed, 2 insertions(+), 1 deletion(-)  
  create mode 100644 anotherfile.md  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master  
$ ls  
anotherfile.md testfile.md  
Shreyas at Jarvis in ~/Documents/_Berkeley/sem2/i290DataMining/workspace on master
```



Git Commands

- Creating New Repo
 - Committing
 - Branching/Merging
 - **Inspection**
 - Cloning a Repo
 - Updating

```
$ git log
```

text-based graph :)

8-1
decorate

[abbey.com](http://www.abbey.com)

11

--all

--pretty-oneline

```
$ git log --graph --decorate --abbrev-commit --all -*
* 1fd3b2d (origin/native-dev) Merge branch 'native'
| \
| * 3a17e61 pointing hand cursor set for close
| * 29fe5e3 filter label made more interactive.
| * d321f17 close button added for search context.
| * a3f3a81 Wit Usage graph and Time saved graph add
| |
| * 2cacb88 Tag search results are listed properly on
| * 6d9e2b3 (origin/qa) Tag filter is working in advna
| * f440164 QA release 0.2.4.0
* 77e05c8 Merge remote-tracking branch 'origin/nat
| |
| * 1843b03 Dashboard wit usage widget improved
| * 737a64e Merge branch 'native-dev' of https://g
| |
| * 66abaf0 side bar made similar to ui concepts
| * f0ea951 fixed issue for coming back from search
| * cfbfa98a Merge branch 'native-dev' of https://
| |
| * 030ff2b dosearch
| * 3e5c533 Top 5 wits graph is implemented
| |
| /|
* 4fed3cb Tags added in wit search result
* 94881c7 First level of advance search integrat
* 61399e3 Merge branch 'native-dev' of https://
```



Git Commands

- Creating New Repo
 - Committing
 - Branching/Merging
 - **Inspection**
 - Cloning a Repo
 - Updating

`~/.gitconfig`

[alias]

```
| = log --pretty=oneline -n 20 --graph
```

```
.gitconfig *  
1 [alias]  
2   # View the SHA, description, and history graph of the latest 20 commits  
3   l = log --pretty=oneline -n 20 --graph  
4  
  
$ git l  
* 883567e70d3325eb566c15bcf743200d8d5a572a Merge branch 'qa'  
* d603345f32fcf4b53325311ccf284fb8756a28d6 Merge branch 'wp-dev' of github.com  
* 1ed9c4d4a0e42a03c95c599baa0d1939db924586 Read file icon association from 0  
* aaa61ad3c87bd0422876971e16decbb0a53363021 Search animation done  
* aa35948ba3bca1560eff28cc89fe90 Merge branch 'wp-dev' of https://  
* daa54c661898a53b2ccdc49e8e09ffbf73fcb8c8 Search bar implemented  
* f6bf2d261d94ac24cc27296d909ff86d6fe15f new api modifications  
* 70ccbbaacd1b3ce7a1e31e1972c119bfb70cd9 search query clear images added
```



Git Commands

- Creating New Repo
- Committing
- Branching/Merging
- **Inspection**
- Cloning a Repo
- Updating

\$ gitk

gitk: Acrowit-Desktop-Widget

SHA1 ID: 40ebc7df1179c2a1f0d82225d0598b78948987 | Row: 16 / 312 |

Find next prev commit containing: Exact All fields

Search

Diff Old version New version Lines of context: 3 Ignore space

Author: Shreyas Chhodas <shreyas.chhodas@gmail.com> Date: Mon Jun 4 09:18:19 2012 -06 -06 09:18:19
Commit: Shreyas Chhodas <shreyas.chhodas@gmail.com> Date: Mon Jun 4 09:18:20 2012 -06 -06 09:18:20
Parent: 5196111e7cc6ca37a34b777fc095a277e534 (4 dot image for drag pattern)
Child: 2169c95c7649ae03c782af84d1993937b06d5 (Merge branch 'wp-dev' of https://github.com:R
Branches: **master**, remotes/origin/dev, remotes/origin/master, remotes/origin/native-dev,
Follows: v0.1.2,0
Precedes: v0.1.2,0

missing text hover for dusk theme added

----- Acrowit-Widget/widgetimages/dusk/icon_32_witdrag_text_hover_itself.png -----
index: 63ddc899..7bb0ef 100644
@ binary files o/Horowitz-Widget/widgetimages/dusk/icon_32_witdrag_text_hover_itself.png.d

Comments: Acrowit-Widget/widgetimages/dusk/icon_32_witdrag_text_hover_itself.png



Working with others...

- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- **Cloning a Repo**
- Updating

github

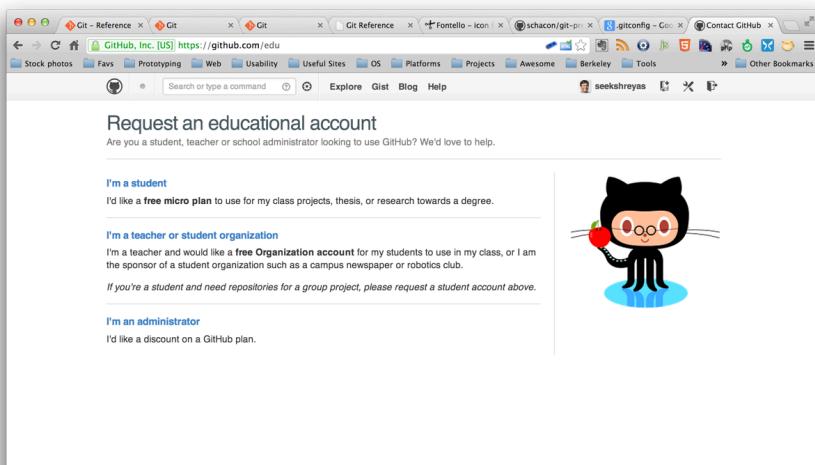




- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- **Cloning a Repo**
- Updating

create an account

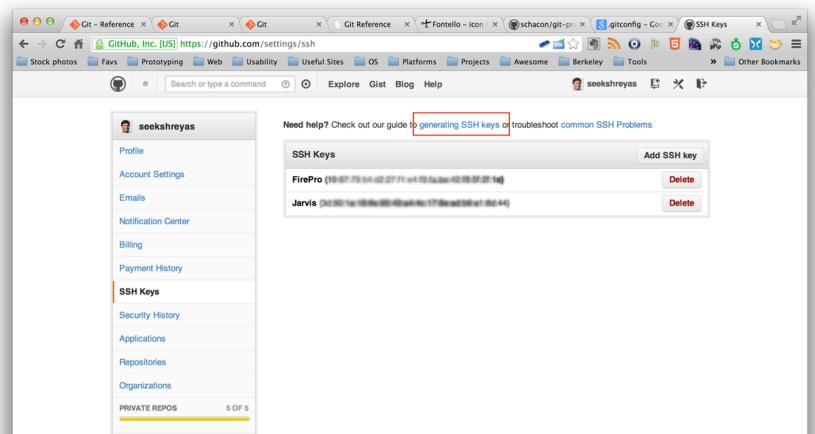
www.github.com/edu



- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- **Cloning a Repo**
- Updating

add ssh key

www.github.com/edu





- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- **Cloning a Repo**
- Updating

\$ git clone <url> <name>

<https://github.com/jblomo/datamining290>

The screenshot shows a GitHub repository page for 'datamining290'. The URL in the address bar is <https://github.com/jblomo/datamining290>. The page displays a list of files and commits. In the top right corner of the repository header, there is a 'Read-Only access' link, which is highlighted with a red box.



- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- **Cloning a Repo**
- Updating

Fork a repo

The screenshot shows a GitHub repository page for 'datamining290'. A modal dialog box is open in the center of the screen, asking 'Where do you want to fork this to?'. It lists two options: 'Fork to seekashreyas' and 'Fork to Acrowit (organization)'. The background of the browser window shows the repository's file list and commit history.



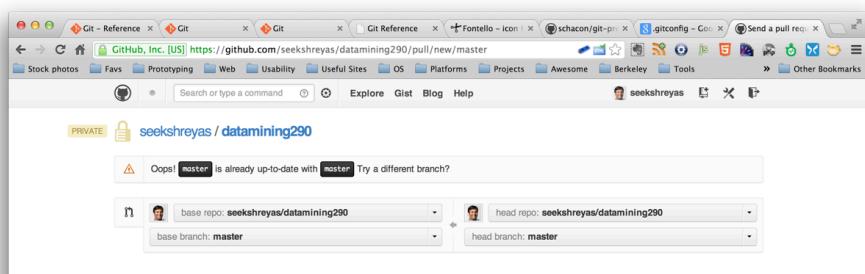
- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- Cloning a Repo
- **Updating**

`$ git pull` = *git fetch + git merge*
`$ git push`



- Creating New Repo
- Committing
- Branching/Merging
- Inspection
- Cloning a Repo
- **Updating**

GitHub Pull Request





GitHub Ninja Features

Markdown

for documentation

Pages

for project hosting

Gist

for project hosting

Service Hooks

'do something' on commit



may the Force be with you

