

# Git and GitHub like a Pro!

(Slides adapted from <https://github.com/mobile-senegal/dem-training/blob/master/DEM-05-CollaborativeTools.pptx> of Thomas Sarlandie by Jan Schoepp, Dr. Christelle Scharff and Dhruv Gandhi - Last update: 4/20/2018)

# Agenda

Git for your projects

Collaboration with Git and GitHub

# **GIT FOR YOUR PROJECT BASICS**

# What is Software Configuration Management (SCM)?

- To manage your source code
- To keep an history of your project
- To facilitate collaboration on a project

# Why you need an SCM

## Source code is a valuable asset

- One of the most important things in a software company!
- You need to **protect** it from
  - Physical damages / Theft of hard drives
  - Human errors

## You will save a lot of time

- An SCM is the best way to synchronize multiple developers without wasting hours *merging*
- It is a great tool when you need to find when a bug was introduced

# SCM History

SCM have been around for decades now

- CVS (1986), Subversion (2000), Mercurial, BitKeeper, Git
- They evolved from *centralized* systems to *distributed* systems

Git is widely recognized as one of the best SCM available today

- and GitHub is the number one place to share code

# Let's get started

## Install Git

- Mac OS: (already installed with Xcode)
- Windows: <http://git-scm.com/download/win>
- `git config --global user.email email` (preferably your GitHub account)
- `git config --global user.name "Firstname Lastname"`

## Create a new project / directory

- `mkdir myproject`
- `cd myproject`
- `git init`

# Add a file

Create a README.md file with some content

- echo This is the readme >> README.md
- git status
- git add README.md
- git commit -m “Adding README.md”
- git status



```
$ git add README.md
$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file> ..." to unstage)
#
#       new file:   README.md
#
```

```
$ git commit -m "Adding README.md"
[master (root-commit) dc95b92] Adding README.md
1 file changed, 1 insertion(+)
create mode 100644 README.md
```

```
$ git status
```

```
# On branch master
```

```
nothing to commit (working directory clean)
```

# Recover from problems

Edit your README.md file again

- Add something “bad” in it

Rollback your change

- `git checkout -- README.md`

```
$ echo "Bad developer joker" >> README.md
$ 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:   README.md
#
no changes added to commit (use "git add" and/or "git commit -a")
$ git diff
diff --git a/README.md b/README.md
index bb69143..91a4a14 100644
--- a/README.md
+++ b/README.md
@@ -1,2 @@
 This is a README.md file for our git exercise
+Bad developer joker
```

```
$ git checkout -- README.md  
$ git status  
# On branch master  
nothing to commit (working directory clean)
```

# Commit a file

Let's edit the README.md file again

And commit the file again

```
$ echo "Good developer joker ;)" >> README.md
$ 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:   README.md
#
no changes added to commit (use "git add" and/or "git commit -a")
$ git add README.md
$ git status
...
$ git commit -m "Adding a good joke"
```

```
$ git log
```

```
commit 7552e81a4a6770532ee52d344e67557c2f50c460
```

```
Author: Thomas Sarlandie <thomas@sarlandie.net>
```

```
Date: Wed May 1 16:59:44 2013 +0000
```

Adding a good joke

```
commit dc95b9254b81db32a2fa16116e7f774eedc3a332
```

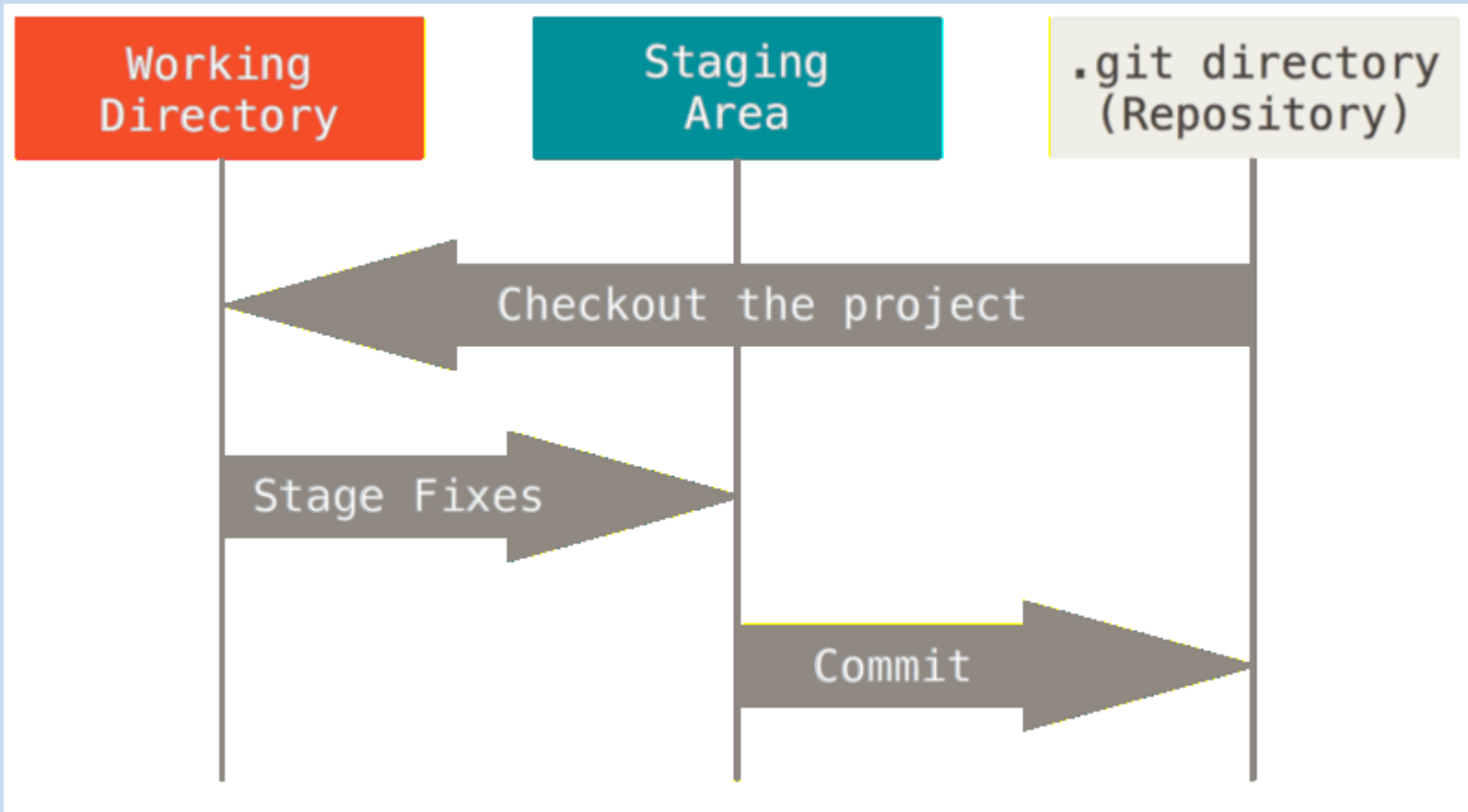
```
Author: Thomas Sarlandie <thomas@sarlandie.net>
```

```
Date: Wed May 1 16:52:54 2013 +0000
```

Adding README.md



# Understanding Git



GitHub Pro book: <http://git-scm.com/book/en/Getting-Started-Git-Basics> - With permission (CC-BY)

# Your essential git basic commands

git init	# Creates a new repository
git diff	# Diff working directory
git checkout -- FILE	# Reverts changes to FILE
git add FILE	# Adds a file to staging
git diff --staged	# Diff staging
git reset -- FILE	# Removes a file from staging
git status files	# Shows you modified/staged
git commit	# Commits staging
git log	# To see the previous commits
git rm FILE	# Remove a file
git mv OLD NEW	# Rename a file

# Basic Git tips

## Getting help

- `git --help`
- `git command --help`
- Google

## Ignoring files

- Add names or patterns to `.gitignore`

## Commit messages

- First line is a short description
- The rest is details of what was done

# **GIT FOR YOUR PROJECT BRANCHES**

# Understanding branches

Branches allow you to keep different version of your project in parallel

- One stable version – one unstable
- Make experiments
  - To try a new library
  - To prototype a feature
- To work with other developers

# Working with branches

```
$ git branch newFeature  
$ git checkout newFeature  
$ git add ... ; git commit  
$ git checkout master  
branch
```

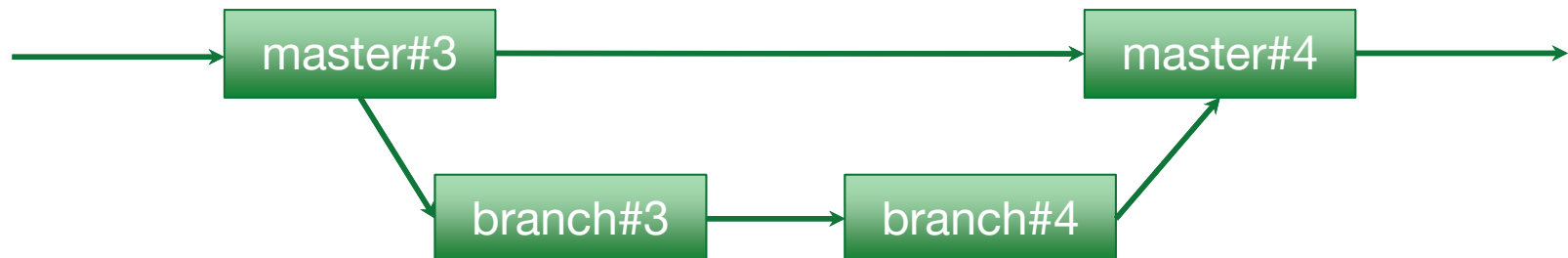
```
# Creates a new branch  
# Switches to the new branch  
# Add changes to the branch  
# Come back to the main
```

```
$ git merge newFeature  
$ git branch -d newFeature
```

```
# Merges the branch into current  
# Deletes a branch
```

```
$ git branch  
branches
```

```
# Lists your
```



# Demo #1 - Branching

Using branches, jQuery and CDN

Compare loading time and merge if necessary

Code at: <https://github.com/mobile-senegal/jquerymobile101>

jQuery Mobile 101

file:///Users/thomas/Desktop/git-exercise/index.html

Google Gmail 29 GCalendar Google Drive Backelite Mail Annotatelt Other Bookmarks

## jQuery Mobile 101

First list item

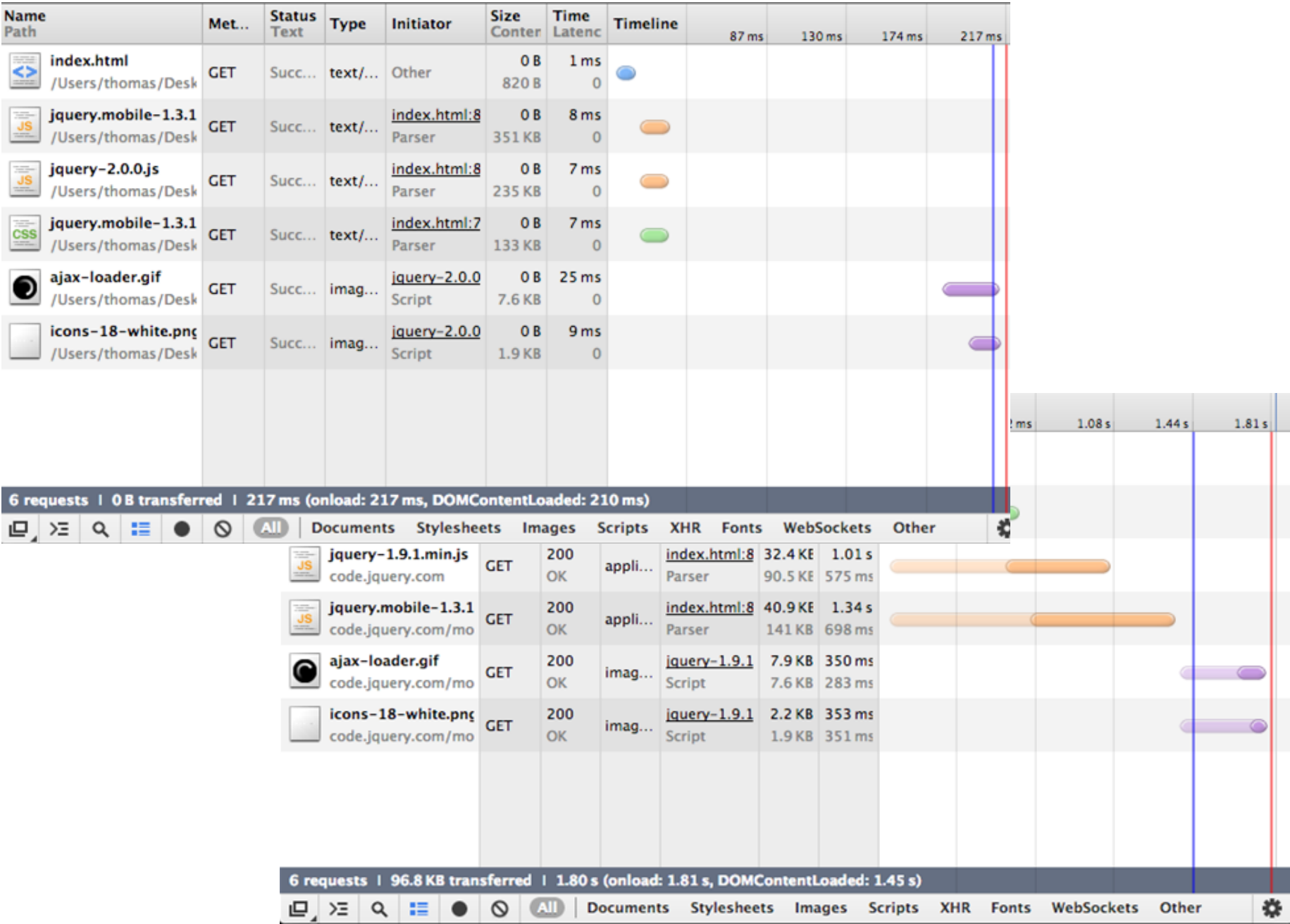
Footer

Name Path	Met...	Status Text	Type	Initiator	Size Conter	Time Latenc	Timeline	87 ms	130 ms	174 ms	217 ms
index.html /Users/thomas/Desk	GET	Succ...	text/...	Other	0 B 820 B	1 ms 0					
jquery.mobile-1.3.1 /Users/thomas/Desk	GET	Succ...	text/...	index.html:8 Parser	0 B 351 KB	8 ms 0					
jquery-2.0.0.js /Users/thomas/Desk	GET	Succ...	text/...	index.html:8 Parser	0 B 235 KB	7 ms 0					
jquery.mobile-1.3.1 /Users/thomas/Desk	GET	Succ...	text/...	index.html:7 Parser	0 B 133 KB	7 ms 0					
ajax-loader.gif /Users/thomas/Desk	GET	Succ...	imag...	jquery-2.0.0 Script	0 B 7.6 KB	25 ms 0					
icons-18-white.png /Users/thomas/Desk	GET	Succ...	imag...	jquery-2.0.0 Script	0 B 1.9 KB	9 ms 0					

6 requests | 0 B transferred | 217 ms (onload: 217 ms, DOMContentLoaded: 210 ms)

All Documents Stylesheets Images Scripts XHR Fonts WebSockets Other





```
$ less index.html
```

```
...  
<title>jQuery Mobile 101</title>  
<<<<<<< HEAD  
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.3.1/jquery.mobile-1.3.1.min.css" />  
  <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>  
  <script src="http://code.jquery.com/mobile/1.3.1/jquery.mobile-1.3.1.min.js"></script>  
=====  
  <link rel="stylesheet" href="jquery.mobile-1.3.1.css">  
  <script src="jquery-1.9.1.min.js"></script>  
  <script src="jquery.mobile-1.3.1.js"></script>  
>>>>>> fixjqueryversion  
</head>  
...
```

This is what a conflict looks like

```
$ vi ...
```

```
<title>jQuery Mobile 101</title>  
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.3.1/jquery.mo  
  <script src="jquery-1.9.1.min.js"></script>  
  <script src="http://code.jquery.com/mobile/1.3.1/jquery.mobile-1.3.1.min.js"></script>  
</head>
```

```
$ git add / git commit ...
```

```
$ git log
```

To fix it: edit the file  
Merge the lines manually  
Remove the markers

# Regrouping

You have all the basics now

- How to work on one branch on your computer
- How to create and work with branches

Now let's do it with friends...

# **COLLABORATION WITH GIT AND GITHUB**



# GitHub

Fork me on GitHub

GitHub is a web service that provides Git repositories

- GitHub for education. Check it out!
- Free service as long as your repo are public
- <https://github.com/pricing>
  - Developer: 7\$ / month for unlimited private repositories (Free for students)
  - Team: 9\$ / user per month for team and user permissions

# GitHub Overview

- To host your projects
  - To track issues
  - To host a web site (static pages only) (GitHub pages)
  - To get analytics on your projects
- 
- To contribute to open source projects
  - To boost your CV and your portfolio

# Remote branches

When working on your computer,

- the entire history is saved in your .git folder

Git allows you to connect to remote repositories and exchange code (push/pull)

- Code is safer when not on your hard-drive...



**Jan Gerhard Schoepp**  
jangerhard

Follow

Block or report user

New York

Overview

Repositories 16

Stars 5

Followers 2

Following 1

### Pinned repositories

#### TwilioEducation

Quizmaster - A project for CS643 at Pace University using Node.js, Heroku, Twilio, and Firebase

JavaScript

#### jangerhard.github.io

Main page for my personal webpage

CSS

#### Heroku-test

Quick testing for Heroku, using a dynamic web project written in Java, with a .jsp file.

CSS 1

#### Beacon-demo

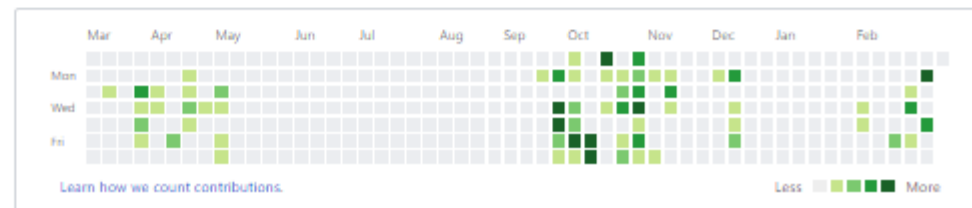
Brief demo to accompany presentation on beacons for Pace University

Java

#### CS6442016


Homework for the class CS644 at Pace University 2016


### 452 contributions in the last year








# Working with GitHub


 jangerhard / TwilioEducation


 Watch 1


 Star 0


 Fork 0


 Code


 Issues 0

 Pull requests 0

 Projects 0






















 Wiki

 Pulse




 Graphs

Branch: master ▾

Commits on Dec 12, 2016

	Update README.md jangerhard committed on GitHub on Dec 12, 2016	 64cef75	
	Update README.md jangerhard committed on GitHub on Dec 12, 2016	 2c830b5	
	Update README.md jangerhard committed on GitHub on Dec 12, 2016	 ff0ee46	
	Fixed small mistake in link jangerhard committed on Dec 12, 2016	 61a8eac	
	Merge branch 'master' of https://github.com/jangerhard/Heroku-node-test jangerhard committed on Dec 12, 2016	 c7af3ea	
	Added link to resume website jangerhard committed on Dec 12, 2016	 d19c844	
	Update README.md jangerhard committed on GitHub on Dec 12, 2016	 82bd24e	

Commits on Nov 2, 2016

	Update README.md jangerhard committed on GitHub on Nov 2, 2016 ✓	 29805c1	
---	---	---	---

# Upload your code

## Create a new repository on GitHub

- Name it whatever you like
- Set to public/private
- **Don't** initialize with a README right now;  
We are importing existing code

## Push to Github

- git remote add origin  
<https://github.com/jangerhard/newAwesomeProject.git>
- git push -u origin master

# Maintaining code

## Before coding

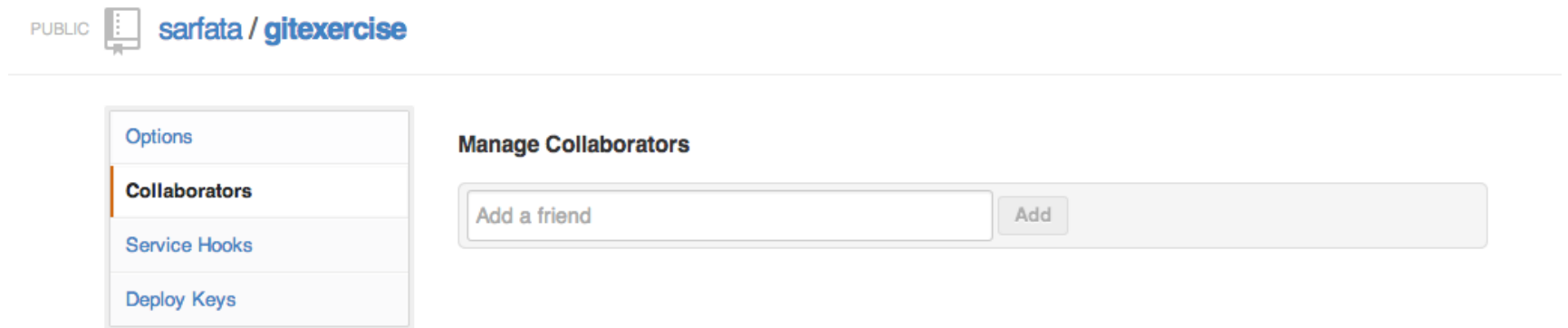
- **ALWAYS** pull latest changes
- git pull origin master

## Before pushing commits

- **ALWAYS** push latest changes
- git push origin master

# Cloning an existing project

Authorize your right-neighbor as a collaborator of your project



Clone the GitHub project of your neighbor

- `git clone http://... *[optional] foldername*`

# **ADVANCED GITHUB PULL-REQUESTS**

# Pull Requests

When you want to contribute to a project but do not have access to it

- Fork the project
  - You will have your own copy of the project in GitHub
- Clone your copy on your computer
- Commit your changes in a branch
- Use the “Pull-Request” button

The author of the project will get a message asking him/her to review those changes

- He can discuss it (comments) and merge it online

# Removing a file from history

When you want to remove a file from the commit history:

- Why
  - credentials/confidential information

```
$ git filter-branch --index-filter "git rm -rf --cached --ignore-unmatch path-to-file" HEAD
```

```
$ git push --all
```

# Summary

You have learned how to ...

- Use Git to be a better developer
- Use GitHub to collaborate





# References

- <https://github.com/mobile-senegal/jquerymobile101>
- <https://git-scm.com/book/en/v2>
- <https://github.com/jangerhard/GitWorkshop>
- <https://github.com/mobile-senegal/dem-training>
- <https://github.com/sarfata>