CS 148 Database Design for the Web

Welcome to the class :)

Robert M. Erickson feel free to call me Bob.
Castleton State University (BS in CIS, 1987)
Clarkson University (MS in MIS, 1991)
UVM ever since.

Syllabus – overview,

What do you expect from this class?

On your own complete: Classroom Activity: Syllabus exercise (BB: Ask a Question)

1

Unix Commands

Command line instructions instead of click with mouse etc.

Longer learning curve but more powerful

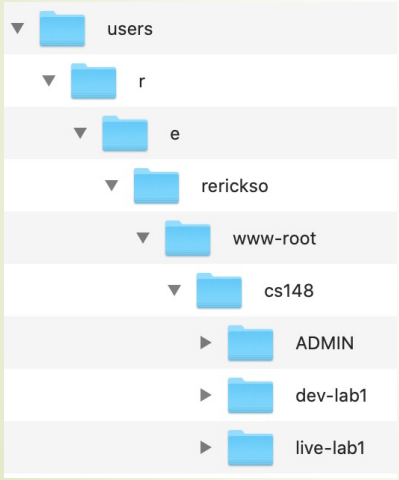
PC – you need git bash shell

Mac you have terminal

Common commands	
ssh w3.uvm.edu –l *your-netid*	Connect to another computer
pwd	Present working directory (folder)
cd *folder-name*	Change directory to this directory
ls	List your files
ls -al	List all the files including system files

2

Unix Commands



unix command: -bash-4.2$ ***pwd***

Will display the path to your current location:

/users/r/e/rerickso/www-root/cs148/live-lab1

The image is what programmers created which is what we are used to seeing.

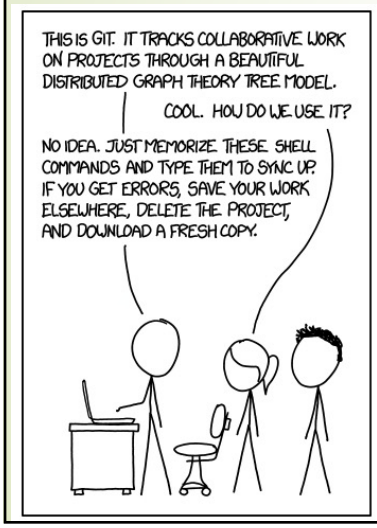
More Unix commands

Self Practice Exercise

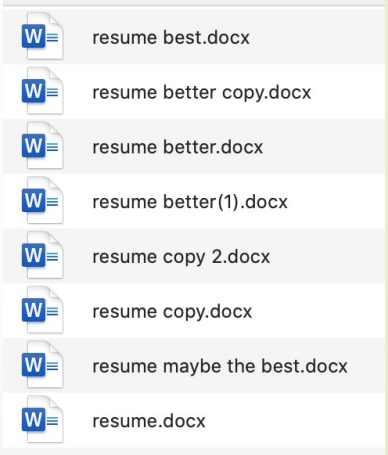
3

Version Control

Keeping track of your changes



Easy way to describe it. Have you ever done something like this with your code? ----->



4

Version Control

Keeping track of your changes

The way to code is to

1. Write a small amount of code.
2. Test it.
3. Save that copy.

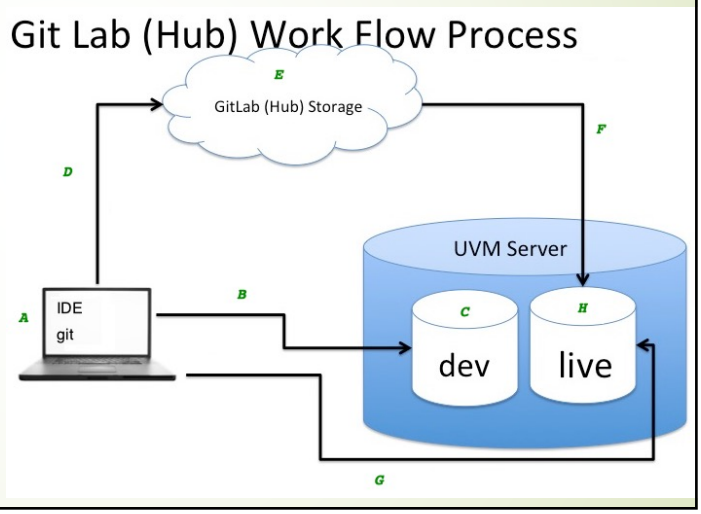
This way mistakes are easier to find

5

Version Control

Git Work Flow - Editing

1. Edit your file in the IDE. (A)
2. sFTP file (A) to dev folder (C)
3. Use Browser to test. (A & C)
4. git status (A)
5. git add –A (A)
6. git commit –m "clear description of task" (A)
7. git push origin master (D)
8. repeat



6

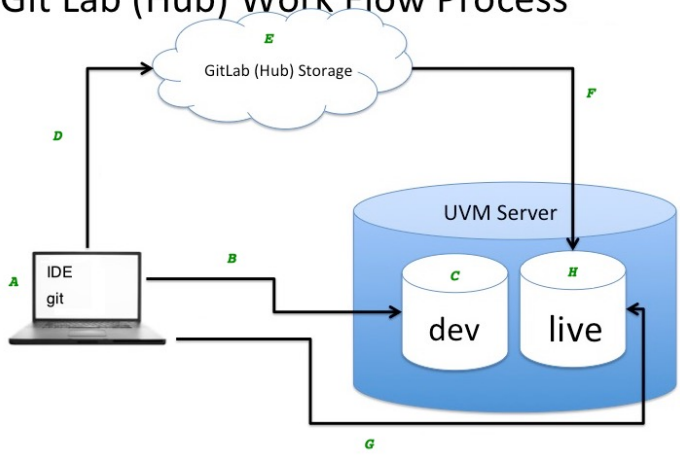
Version Control

Git Work Flow – Going Live

Ready for this version to go live:

8. ssh w3.uvm.edu –l netid (A)
9. cd www-root/cs148/live-lab1 (B)
10. git pull origin master (B)
11. Use browser to verify live site is working (A & H)
12. Repeat as needed

Git Lab (Hub) Work Flow Process



7

Git -> GitHub setup

Set up steps on your computer

1. Create your folder: dev-lab1
2. Get to a unix terminal on your computer
3. cd to your dev-lab1 folder
4. Initialize the folder to start git (no files are needed): git init
5. Connect to your GitHub account: git remote add origin (ssh string)
6. Set up your .gitignore and other files
7. Generate the SSH key for your computer
8. Copy the ssh key to GitHub

8

Git -> GitHub setup

Work flow steps on your computer

1. Edit your files in your IDE as you would normally
2. sFTP your files to the dev folder making sure your code works
3. When you have created good working code:
4. Go to the terminal on your computer
5. cd to your folder
6. git status. [verify the files are expected, if not …]
7. git add –A [to add all of them or git add folder/file]
8. git commit –m "description of what you worked on"
9. git push origin master. [sends copy to GitHub, backup]
10. repeat 6, 7, 8, 9 as often as needed (dozen times plus a day)

9

Git -> GitHub setup

Set up steps for your silk account

1. Use terminal to connect to server: ssh w3.uvm.edu –l netid
2. cd to your class folder www-root/cs148
3. Create your live folder: live-lab1
4. Initialize the folder to start git (no files are needed): git init
5. Connect to your GitHub account: git remote add origin (ssh string)
6. Generate the SSH key for your uvm account
7. Copy the ssh key to GitHub (you will now have two keys in GitHub)
8. Git pull origin master
9. Repeat Steps 1, 2, 8 as needed to go live.

10

Take away

The way to code is to

code little pieces,

check to make sure it works.

Commit that working code

Repeat

Pull live as needed

If you mess up …. Have patience and be sure to ask. Practice - Comfort

11

Basic CSS

defines the look of your web site

selector

```
body{
    background: url("banner.png");
    font-family: Arial, Helvetica, sans-serif;
    margin: auto;
    width: 90%;
}
```

Declaration block

value

property

```
background: url("../images/banner.png");
```

12

Flex Layout

Its all about parent and children

Flex Container is the Parent: display: flex

Flex Item is the child: flex: grow, shrink, basis;

flex: 0 1 20%;	will be 20% of the screen (max size) but can shrink smaller
flex: 1 0 20%	will be 20% of the screen (min size) but can grow larger

13

Grid Layout

Its all about parent and children

display: grid; **is the Parent**

grid-area: *name-you-give-it*; **is the Child**

grid-template-area: "nav article side"; **is the "cells"**

defines how many columns you have
putting a name in twice allows that name to span two columns

14

Questions ?

15

15