Internet and Cloud Systems

http://thefengs.com/wuchang/courses/cs410c/
About the course

- A tour of systems and the abstractions they provide
  - Historical look at where we have come from
  - A survey of modern systems
- Hands-on practice using them to build applications
- First offering (apologies in advance!)
About the course

- Intended as an overview of our electives, but with a cloud-focus
- Coverage of:
  - Internet and its protocols
    - CS 494/594
  - Web development
    - CS 410/510: Full-stack Web Development
  - Databases
    - CS 486
  - Virtual machines and containers
    - CS 533
  - Security
    - CS 491/591
  - Data science
    - CS 410/510: Cloud & Cluster Data Management
  - Machine learning
    - CS 445/545
- Will be some review if you’ve already taken the elective
Course objectives

- Understand Internet networking and how it has enabled the cloud.
- Perform web development and construct web applications backed by databases.
- Utilize virtual machines and containers as building blocks for constructing services.
- Adapt applications to models of computation and storage supported by cloud computing providers.
- Utilize cloud tools for data analysis and machine learning applications
Assumptions

- Working knowledge of Linux (or willingness to do a crash course in learning it)
  - http://linuxcommand.org/lc3_learning_the_shell.php
  - http://overthewire.org/wargames/bandit/
- Knowledge of processes and operating systems (CS 201)
- Some knowledge of HTML (or willingness to watch a screencast or two to acquire it)
- Some knowledge of Python (or willingness to do work outside of class to learn it)
Format

- Lectures, labs, homework and projects

Learning Pyramid

- Lecture: 10%
- Reading: 20%
- Audiovisual: 30%
- Demonstration: 50%
- Discussion: 75%
- Practice doing: 90%
- Teach others: average student retention rates

Source: National Training Laboratories, Bethel, Maine
Grading

- Attendance graded
- Participation graded
  - In class and on channel
  - Pointing out (the inevitable) mistakes, fixing bugs, suggestions, troubleshooting problems for others
- Labs done in pairs or individually
  - Lab notebook to document completion due by final exam slot (Monday of Finals week)
- Homeworks done individually
  - Code and container submissions
  - Due a week after assigned on Bitbucket or Docker Hub repo
- Final project done in pairs (can be different than lab group)
  - Due as a screencast demo and walkthrough at the end of Finals week
- Percentages, rubric, and schedule on course page
Homework #1: Slack, Docker Hub

- Slack
  - Create an account and join the Slack channel at [https://pdx-cs.slack.com](https://pdx-cs.slack.com) (#cs410_510_cloud)
  - All course communications through Slack
    - DM @wuchang
    - Anonymously message me via `/abot @wuchang <msg>`
    - Can also use [https://sayat.me/wu4f](https://sayat.me/wu4f) for anonymous feedback
- Docker Hub ([https://hub.docker.com](https://hub.docker.com))
  - Create a Docker Hub account
  - Container submission will be via upload to Docker Hub
Homework #1: Bitbucket

- Create a Bitbucket account if you don't have one at https://bitbucket.org
- Find the + and click on it to create a new repository
  - Name your repository cs410c-<OdinId> (e.g. cs410c-wuchang)
  - Keep the default settings then "Create repository"
- Pause at the instructions for "I'm starting from scratch"
- Set-up ssh key access from linuxlab.cs.pdx.edu
  - If you have not set-up an ssh key before, log into a linuxlab machine and run: ssh-keygen -t rsa
  - Copy the contents of ~/.ssh/id_rsa.pub to the clipboard
  - Go back to Bitbucket to use that ssh key to access your account (for convenience)
    - https://confluence.atlassian.com/bitbucket/set-up-an-ssh-key-728138079.html
- Then, follow the instructions for "I'm starting from scratch" to commit a README.md file in the top-level of your new repository
- Go back to the new repository on Bitbucket and find "Send invitation"
  - Add course instructor with Read access (wuchangfeng)
Homework #1: Bitbucket

- Read the first 6 steps of [https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners](https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners)
  - `git clone`
    - Fetch a copy of a remote repository
  - `git add`
    - Add a new file and/or directory to local repository
  - `git commit`
    - Commit changes to local repository
  - `git push`
    - Merge changes from local repository to a remote one
    - Implicitly assumes "origin" (place that you retrieved repo from) and "master" (branch)
  - `git pull`
    - Merge changes from remote repository to your local one
    - Implicitly assumes "origin" (place that you retrieved repo from) and "master" (branch)
Homework #1: Bitbucket

- Add to your repository and push changes to BitBucket
  - Create a directory called `hw1` within local repository (`mkdir hw1`)
  - Within the directory
    - Create a markdown file called `README.md` that contains a colorful message to me that shows me how much markdown you know
      - `cd hw1; vim README.md`
    - Create a file called `dockerhub.txt` that contains a link to your Docker Hub repository
      - `vim dockerhub.txt`
  - Add the directory and files to your repo via `git add .`
  - Commit the changes locally via `git commit`
  - Push the changes to your remote BitBucket repo via `git push`
- Other resources
  - [https://confluence.atlassian.com/bitbucketserver/basic-git-commands-776639767.html](https://confluence.atlassian.com/bitbucketserver/basic-git-commands-776639767.html)
Homework #1: Bitbucket

- Note: All code submissions will be via Bitbucket
  - Commit changes to your repository frequently
  - Use descriptive messages in commit messages
  - Create a tag for your repository for each homework submitted
    - Tags named hw1, hw2, hw3, … etc.
Homework #1: Google Cloud Platform

• Google Cloud Platform (GCP)
  • Log out of your personal Google account
  • Go to https://console.cloud.google.com and login using your pdx.edu account to enable GCP
  • If you haven't used GCP yet and you do not mind temporarily putting your CC on the account, apply for the $300 coupon and use it to create a new billing account
    • https://codelabs.developers.google.com/codelabs/cpb100-free-trial
  • Or, wait for me to e-mail you the course coupon, then apply it and create the course project
Homework #1: Google Cloud Platform

- Create a project called cs410c-<OdinId> (e.g. cs410c-wuchang) to ensure your OdinID appears in all screenshots for your lab notebook
- Associate the billing account to it
Homework #1: Google Cloud Platform

- Throughout the term, to get a feel for what costs money, check detailed billing.
Homework #1: Local VM setup

• Install a Ubuntu 16.04 VM with Docker on the workstation you will be using for the quarter
  • [http://thefengs.com/wuchang/courses/ubuntu_virtualbox.txt](http://thefengs.com/wuchang/courses/ubuntu_virtualbox.txt)
  • Be sure to name your VM with your OdinID so that it appears in all screenshots for your lab notebook
Homework #1: Local VM setup

- Insert Guest Additions CD and install them within VM
- Then, power down the VM
Homework #1: Local VM setup

- Tweak VirtualBox General settings for your VM to enable shared clipboard
Homework #1: Local VM setup

- Tweak VirtualBox Display settings for your VM to improve graphics performance (128MB memory and 3D acceleration)
Homework #1: Local VM setup

- Ensure VT-x is enabled

[Image of a settings window with a checkbox for enabling VT-x and nested paging]
Homework #1: Local VM setup

- Ensure at least 2048MB of RAM is allocated
Homework #1: Local VM setup

- Launch VM and within it
  - Set up another ssh key to access your Bitbucket account from your local VM
- Then, clone the course's source repository as well as your own
  
git clone ssh://git@bitbucket.org/wuchangfeng/cs410c-src.git
git clone ssh://git@bitbucket.org/<bitbucket_id>/cs410c-<OdinId>
Homework #1: Lab groups

- Default groupings done alphabetically
  - If you would like to pair up with someone specific or work individually, let me know at the end of class