

## ICD571: Network Workflow (1 credit, v0.1)

### Course Information

- URL: tk
- Instructor: John Bell, email: john.bell@umit.maine.edu skype: nmdjohn
- Synchronous Hours: tk

### Course Description

Students will learn the basic computing systems and conventions underlying networked, web, and asynchronous project production. Before digital creators make their work public or collaborate with others across a network they must first master the tools necessary to connect to servers, move files, maintain versions, and otherwise operate in a digital workplace. This class will focus on standard web environments like Unix and Linux servers, but the clients used to connect to those servers will be standard Windows and OS X programs.

### Course Goals

Upon completing the course, students will have:

1. Been introduced to Linux-based command line operations, permissions, and file versioning tools.
2. Been exposed to patterns used in setting up the file structure of a web site.
3. Developed an understanding of file operations and basic security.
4. Prepared to publish content in a network environment.

### Required Materials

*Required Reading:*

- none (online references will be assigned each week)

*Recommended Reading* (relevant, but not used in the class):

- none

### Course Flow

The course will consist of a series of video lectures and associated assignments broken up into four week-long topics, with an extra week to finish all your assignments and revisions. Within each week you will be expected to:

- Watch all of the lectures and tutorials listed in the week's introduction and read any assigned material.
- Submit the weekly assignment.
- When relevant, add comments to the submissions of at least three other students. You will be provided with specific aspects of the assignments to focus on when critiquing your classmates' work. The nature of this class means that this objective will not be part of the assignment every week.

This course is designed to be completely asynchronous so there are no specific times for meeting with other students or the instructor. Instead, interaction will take place via email and the course's message board, where you will also be submitting assignments and giving feedback to other students.

### **Attendance Policy**

Attendance in an asynchronous online course is a somewhat nebulous concept. While it is expected that you will accomplish all of the tasks by their assigned deadlines, participation in the class' online discussion is also critical to your success in the class and the frequency and depth of your interactions with other students and the instructor will be considered part of your "attendance" and thus part of your grade. **If you for any reason think you may have an issue, either on a specific day/week or overall, talk to us! It is much easier to make accommodations ahead of time than after the fact.**

### **University Policies**

**Disabilities:** If you have a disability which will make it difficult to meet the expectations of the course, please contact Ann Smith, Director of Disability Support Services (581-2319), as soon as possible.

**Academic Integrity:** Academic honesty is very important. Copying and plagiarism are unacceptable. As described in the student handbook, violations of the University's academic honesty policy can result in consequences ranging from failure on the assignment to failure of the course and referral to the University for disciplinary action.

### **Participation**

If the only time you post a message is when you're turning in an assignment then you will have little opportunity to display your understanding of the ideas being discussed and I will not have much information to use when evaluating your success in the class. Ask questions, throw in comments, and generally add to the discussion as much as possible, *particularly* if you think you missed something or you have a stupid question. Odds are other people are as confused as you are.

As with all classes, it is expected that you will treat others with respect. If you are repeatedly abusive toward your classmates you will be asked to leave and the day will be considered an absence for purposes of the attendance policy.

### **Grading**

Grading for your assignments is weighted equally across each of the five weeks of the class. You will be primarily graded on the same criteria you use to give feedback to other students each week.

Participation is a significant part of your grade. The more you add thoughtful, insightful comments to the discussion the more both you and other students will benefit. Questions are always welcome and should be asked publicly so that everybody can see the answer unless there is a very good reason to ask privately. Your critiques of other students' work

are considered an absolute minimum level of participation that you should rise above as much as possible.

After submitting your assignments you will get feedback from the instructor and at least 3 other students. You may, and in fact probably should, revise and resubmit your assignments after it is critiqued: the goal of doing an assignment is to learn, and learning requires applying feedback instead of just moving on to the next task. Your assignment will not receive a final grade until you decide it is done, though the instructor reserves the right to apply diminishing returns to multiple submissions.

## **Phases**

### *Phase 1: Making connections*

Before you can put files on a server you first have to connect to it. The first phase shows you the different types of connections you can make to a server, common software used to make those connections, the types of accounts necessary, and the security implications of each protocol.

### *Phase 2: Navigating a Unix-style file system*

Once you connect to a server you'll need to know where to go and what to do. \*nix file systems are very different than Windows and though OS X is similar it tends to insulate users from some of the trickier concepts. This phase shows you standard locations for files and where you should and shouldn't go when logged into a \*nix box.

### *Phase 3: Playing well with others*

Programmers tend to be particular about things like file names, capitalization, and spelling for good reason – if any of those things are wrong even once in thousands of lines of code a program won't run properly. In the final phase you'll learn some of the conventions used in programming so that you can both apply them to your own files and recognize them when you want to use somebody else's code. You will also be introduced to versioning and some of the software used to manage file versions, a critical tool when working on code with other creators.

## **Timeline**

### *Phase 1: Making connections*

- Connecting to a server
  - Learn the methods, protocols, and software used to connect to a web server in different ways.
  - Videos: Clients and servers, CLI, TCP/IP, Telnet, SSH, S/FTP, WebDAV, HTTP/S
  - Connect to the server using the parameters and credentials provided in the assignment. You'll have to do certain operations on the server to demonstrate that you've connected successfully.
  - Check to make sure that your classmate's files are in the right place.

### *Phase 2: Navigating a Unix-like file system*

- Welcome to \*nix

- Get the lay of the land in \*nix-world, including basic commands, concepts, and standard file locations.
- Videos: Creating, moving, and deleting files, pathnames, standard paths, filename conventions, permissions, Telnet and FTP
- Complete the given list of operations on the server.
- Check to make sure that your classmate's files are in the right place and have the proper permissions set
- Putting \*nix to work
  - Now that you can move around in \*nix, you can start to see how to use it to publish your content to the web or share it with your co-workers.
  - Videos: File security, file editing, users, groups, processes, cron, Apache, passwords, intro to scripting.
  - Complete the given list of operations on your web site.
  - Connect to your classmate's web site and verify that it is working.

### *Phase 3: Playing well with others*

- Languages, names, and letters, versioning
  - Learn the conventions programmers use when they are writing code and building web sites and how to make simple tweaks to found code.
  - Videos: File naming, path conventions, common web languages, variables and constants, strings, readme, APIs, text conventions.
  - Customize the provided software package and install it on your web site.
  - Check on your classmate's web site to ensure they have installed their software correctly. Make sure that it both operates properly under normal circumstances and that it correctly handles situations where the user doesn't behave as expected.
- Versioning
  - When working on a team it is important to make sure that your work doesn't accidentally collide with the work of others, breaking it. Versioning software is how programmers do it.
  - Videos: Wikipedia, file versions, view source, Git, Subversion, and CVS, versioning clients, checkout and commit, change logs, comments
  - Create a Git repository, connect to it with your client, and create a head revision of your files.
  - Check out, modify, and commit files from your classmate's repository using the connection info they provide.