Micah's IU Stuff 

Micah's Stuff F401 R341 W220

W220 Syllabus

W220: Tech. Issues:Computer-Based Education

Instructor: Micah Gideon Modell
Office Hours: School of Education Rm 2010, Thursdays from 4pm to 5pm or by appointment

Introduction

Students in this blended class will understand more than just how to write a small but useful program. They will teach each other the history of computing, beginning with the hardware (which they will assemble), operating systems (which they will install) and, ultimately, the evolution of programming and programming languages (which they will work with). This will all contribute to their ability to apply this knowledge and implement an efficient team-based software development lifecycle. In effect, they will be well prepared to support technology integration within their school systems and to teach their own students to be similarly competent.

Assuming you want *they* to be *you*, this document is for *you*.

Objectives

Students successfully completing this course will demonstrate the ability to:

- Teach computer science topics
- Modify existing computer program source code to improve and/or change what it does
- Contribute to a collaborative team effort

Some of the specific tasks include:

- Actively participate in online, forums-based discussions
- Structure and lead an online, forums-based discussion
- Consistently apply a fair set of grading rules to participation in online, forums-based discussions
- Explain the origins of modern computing
- Explain computing concepts
- Identify the parts of a computer and describe their purpose
- Modify the functionality of a software component by modifying the code itself, including:

- Flow control
- Variables
- Functions
- Object oriented programming
- Databases

Apply software development 'best practices'

Course Structure

Group Work:

Everyone will be assigned to a group for the purposes of completing the computing-based assignments. These groups will be permanent so you should get to know each other quickly, know each others' schedules and learn how best to work together. The group tasks will be graded as a group with possible adjustments to be applied at the end of the semester based upon continuous assessment reports (see Grading below).

In Class:

This class meets every Wednesday from 5pm to 9pm in ED-2025.

Each class will begin with a class activity designed to help you understand the programming challenge for the week. With the activity completed and discussed, we will present the programming challenge itself and we will look for and try to understand documents which might help us to complete these tasks. We will then break off into our groups to plan and begin execution of the task.

Outside Class:

This class includes a mandatory online participation component which will culminate in a quiz at the end of each week. Each week's discussion and quiz will be facilitated in the OnCourse forums component. The results of these quizzes contribute substantially to your grade in this course.

The first of these will be facilitated by me, but subsequent weeks will be run by you (specifically, one of you each week). A schedule will be posted in OnCourse and the schedule below will give you an idea of what your topic will be — check it ASAP so you can begin to prepare (please see Course Materials below for guidance)! Before your week begins you will also receive some further hints as to related content to expect on the quiz (not a complete set, but some general hints). The class average will serve as a multiplier for bonus points awarded to the student instructor for the week (see Grading below).

Grading

The three major components of this course are:

- the online discussions (40%)
- the group programming projects (40%)
- the teaching (20%)

They each comprise a portion of your grade. The grading of each component is described below. Please do not hesitate to contact me with any questions (after this has been explained in class):

The Programming Projects

Each group will build, install and manage their own server machine for the purposes of this class. This is where I will look for the assignments when it comes time to grade them. I will only grade live code that has been published on the server (and we will go over what is meant by this in class). Each project will be graded and every member of the group will receive that same grade for the project.

Every week, each student in the class must complete an online evaluation of every student in their group (including themselves). This will be done using the Continuous Assessment online tool (accessible through OnCourse) and is a requirement. These evaluations will be reviewed during the course for trends in group performance . If I see an issue (e.g. a few weeks of low ratings), I will contact you to see how we can resolve it and get back on track. If problems persist and Continuous Assessment indicates that one or more students are consistently doing more than their share of the work, points for the overall Programming Projects will be reallocated within the group accordingly. It's OK to have a bad week and make it up later, but you will find it tough to make up for a semester of slacking).

Bonus points are also available per project. I look to you to be creative and differentiate your work. You will be rewarded for doing so.

The Online Discussions

Each week, every student is expected to participate in the online discussions. This means following the rules set out by the facilitator for that week points will be awarded to each poster each week by the facilitator (student-facilitator awards will be reviewed and verified). This will comprise the participation component.

A portion of this grade will be driven by your performance on the weekly quizzes. These quizzes will be norm-referenced to adjust for discrepencies in teaching ability.

You are required to be in class for *every* class period. If you will be absent for any class period, notify beforehand so we can work out an alternative. If an emergency arises, notify me as soon as possible so we can work out an alternative. Unexcused absences will result in points being deducted from the Discussions portion of your grade.

The Teaching (bonus)

When it is your turn to teach, you will be expected to seed and nurture the conversation in the forums. At the end of the week, you will be expected to allocate a set number of points amongst your classmates based upon their participation. This allocation will be viewed as strong guidance with respect to the week's participation grades. You will receive feedback on the allocation and on the instruction itself in the form of a completed rubric (in OnCourse).

In addition, each week's average class quiz performance will indicate how many *bonus:good teaching points* you will be awarded.

Bonus points may also be available here if one offers specific quiz questions or prepares and facilitates an in-class activity. Both of these possibilities require at least one week of advanced notice to the instructor to arrange the details.

Course Materials

This course is designed to empower its students to be self-sufficient continuous self-learners and therefore all resources required for in-class projects will be obtained through web searches. These will be performed live in class and/or through the "Pencasts" section of this site (powered by LiveScribe).

However, your outside class activities will require research and you may want a greater level of depth on the in-class activities than this course can directly provide (Linux, subversion, apache, etc.). Fortunately, IU partnerships provide us with access to numerous resources for such purposes and I particularly recommend:

The resources on the UITS Self-Study Training page, including:
- Lynda.com
- STEPS Workshops

The IU Libraries' Resources, including electronic books from:
- Books24x7 and Books24x7 ITPro
- netLibrary

I found value in the …*For Dummies* series of books when I was starting out with unix years ago. I cannot vouch for them today, but they would certainly be a good place to start (i.e. Linux for Dummies). Other resources such as operating systems and development tools will take advantage of a mix of free and open source software as well as software and hardware provided by IU.

Academic Misconduct

The class is morally and procedurally bound by IU's policies on academic misconduct, the details of which you can read about at the following website: http://www.indiana.edu/~code/code/index.shtml

In this class, if you are able to gain access to our machine without rebooting it, physically tampering with it or compromising access in any way, you may earn bonus points by bringing this to our attention. You will be asked to explain the actions you took and how you would guard against them in the future and are free to take advantage of any insights gained. However, *this does not imply license to damage the machine (or its files) or to copy the files on it and pass them off as your own work product — such actions would definitely be viewed as academic misconduct.*

Teaching & Learning Philosophy

Learning is an inherently situated and social activity. As your instructor it is my job to make information and techniques available to you, to give you realistic problems that challenge you to apply that information and those techniques and then to give opportunities for reflection and feedback on what you've accomplished and how. Much of your work will be as part of a team because it is by pooling your knowledge and helping each other that you will all gain a deeper understanding of the subject and because team work is a crucial part of professional life.

There is also a special place in my class for creativity. It is by exploring the limits of your knowledge, challenging your understanding and experimenting with new ways of doing things that you will develop your own interest in the topic. Giving extra credit to those who go beyond the content is one of my favorite things as an instructor.

Week	Programming Task	New In-Class Concepts	Discussion Topic
1 1/11/12	Introduction to the Course		Disk vs. RAM
2 1/18/12	Build your server	Hardware	
3 1/25/12	Install your server software	Operating Systems	My OS is better than yours…
4 2/1/12	Install and Test your Plugin	Software	Master and Servant
5 2/8/12	Improve upon your Moodle Block	php, variables, if statements	It's a string! It's an integer! No, it's a variant!
6 2/15/12	Improve upon your Moodle Block	Loops, conditions and arrays	Controlling the Flow
7 2/22/12		Functions and Finding Information	File Formats
8 2/29/12		Documenting and Source Control	Not Invented Here
9 3/7/12	Install Poll Block	User Roles	Multi-tasking vs. Multi-threading
Spring Break			
10 3/21/12	Object Orientation	Object Oriented Programming and Refactoring	My language is better than yours…
11 3/28/12	Databases	Database Tables and SQL	Of DBAs, SysAdmins and Architects
12 4/4/12			Deployment Models
13 4/11/12	Implement Security Library	Security, Encryption	Top Secret!
14 4/18/12	Administration Pages and		The Life of Software

15 4/25/12	Realtime Polling	Your Choice	

Questions? Comments? Concerns? [mail me]
Follow me on Twitter