Synthesis and Application Essay #2
Barry Peddycord III

Introduction to Teaching	*Effective Questioning Strategies*	*Learning Styles*
Dr. Barbi Honeycutt	Dr. Barbi Honeycutt	Dr. Barbi Honeycutt
August 22, 2011	June 25, 2012	September 26, 2011
3118 Talley Student Center	Online	230 Research Building III

When I started my graduate studies, I was immediately placed into a research position. While I certainly appreciated the security of this situation, I had always wanted to make teaching a major part of my time while pursuing the Ph.D. I managed to make the most of my situation by attending the Fundamentals in Teaching workshops offered through the Graduate School. Even though I wouldn't really get the chance to teach until a year later, I wanted to be as prepared as possible for when that time came. I made attending these workshops a top priority, completing almost all of the major workshops offered through the program, exploring topics such as Learning Styles, Evaluation and Grading, Effective Questioning, and so on - some of them more than once.

The first workshop that most students attend when participating in FIT workshops is the "Introduction to Teaching", a crash-course on basic elements of teaching from Bloom's Taxonomy to designing a lesson plan. I was no exception, and I've actually taken the introductory workshop each semester since Fall of 2011. As a naturally curious academic, it's easy to get off track when going down the rabbit hole of research papers and conference presentations. The Introduction to Teaching workshop gives me a chance to get “back to basics” and realign all of the things I've explored with fundamental concepts in learner-centered teaching. It was especially interesting to attend this workshop in Fall 2012 after I had actually been teaching my lab for two weeks. We were no longer discussing abstract issues that I *might* have to deal with at some unspecified point in the future, but issues that I *had* dealt with first hand in just my first few weeks of class.

During the "Introduction to Teaching", our guiding exercise is to develop a lesson plan for a single day of class. In the Fall 2012 workshop, I actually had the chance to write a lesson plan that I would actually use. That plan ended up being so useful, that I decided to go ahead and make a lesson plan for each and every lab I taught. The lesson plan is still heavily based on the blueprint provided in the workshop, though I've made modest modifications here and there as I develop my own teaching flow. Each lesson plan covers the topics I want the students to learn, how I'm going to deliver the information, how I plan to assess that learning, and the technical considerations I need to prepare for before the lab is held. Since I teach once a week, this gives me plenty of time to prepare before each meeting of the class. While many of the lab activities are outside of my control, I still feel as though I have the responsibility to make the purpose behind each one explicit for the students, and the lesson plan reinforces that focus.

However, the real value in the lesson plans doesn't take shape until after the labs are over. After each session, I do a post-mortem analysis of the lab. Each lesson plan ends with questions used to reflect on what worked and what didn't work, how well I stuck to the original plan, and major issues that students had with the lab session. This encourages me to learn from my mistakes and inform the development of future lesson plans. While a lesson plan could certainly stand on its own as a means of preparing for

labs in the future, this makes them valuable as reflections on labs held in the past. My intention is to include these lesson plans in my teaching portfolio, and taking the time to reflect on my lab sessions makes these into extremely useful artifacts detailing my growth as an educator.

Nearly every FIT workshop has something that can be applied to the planning of class sessions, and several have influenced the way I write my lesson plans. Since this is a lab section, many of the exercises are designed for active learners. Because I write the broader themes into the lesson plan and use those to drive post-exercise discussions, this gives reflective learners the same opportunity to learn and participate as the those who jump right into the activities. I write my lesson plans are written from a global learner's perspective, focusing on broad themes and connections, serving as a framework for the more detailed sequence of steps that will be illustrated in the lab slides. The workshop in Effective Questioning Techniques, on the other hand, has helped me in scaffolding both my questions and the exercises I use to get students to think at higher levels of abstraction. After each exercise, I ask students questions in an attempt to get them to connect the concepts we cover in the lab to the concepts covered in the lecture under the lead instructor's control. Labs are only held once a week, and the lab is used at the very beginning of the assignment, before the student has had a chance to get confused by it. I use this opportunity to give them smaller bits of exposure to course technology in the controlled environment of the lab so that when they tried it for themselves at home, they knew what a “successful” result looked like, and could recognize when they reached it.

What follows is the lesson plan and post-mortem for my first lab of Spring 2013. In Fall 2012, due to when the semester started, students attended the lab before ever attending a lecture, but this semester, they had two lectures before the lab. However, in both cases, the first lab is very important as I have to convey the massiveness of the lab portion of this class, something that many students aren't prepared for when they're just getting started. This is a good glimpse at how I use the first lab to reinforce the importance of the lab exercises for the students – in fact, most of why it is one of my favorite lesson plans is because it reflects back to the chaos of the same lab from 2012 (delivered before I started making lesson plans). In the lab, for example, I slowly build up the complexity of installing the class code base so that instead of immediately having them install it on their own machine, they do so on the lab computers where half of the steps have already been completed first. Unlike last semester, where being unable to install the software stopped most students from getting the first homework assignment done, this semester, more students completed the assignment with fewer questions, and more students took advantage of the lab computers that they were familiar with from our lab section.

One of the challenges with making a lesson plan for a laboratory course is that most of the early labs are designed in advance, and many of the labs later in the semester are scheduled as “working days”. In both cases, I feel out of the loop and without much control. However, in spite of this, I still make it an effort to write a lesson plan for the lesson I'm instructed to teach. I am always thinking about what the learning objectives for the course are and how to best prepare my students to complete their final projects, and doing my best to write the lesson plan from a learner's perspective. During labs, I often have to give ad-hoc mini-lectures on the technology we use, and by keeping track of those in my lesson plan, I can turn those ad-hoc lectures from the first semester into intentional full-blown classes in the second. I started seeing working days for students as “free days” for me to cover whatever material I thought was necessary to help students prepare for their major assignments. If you give students 2 hours of unstructured working time, they are eventually going to drift off task, so by putting together an exercise, I'm able to direct some of that energy towards a concrete goal.

It's a lot of work to put together these lesson plans each day, especially without being required to, but I'm confident that it made up a large part of my successful teaching last semester.

CSC 326: Software Engineering
Lab 1 Lesson Plan

Barry Peddycord III

10 January 2013

Overview and Purpose

Lesson Topic

Welcome to Class!

Learning Outcomes

By the end of this lesson, students should be able to...

- Understand the purpose of CSC 326 (not about programming, about people and process)
- Recognize the key concepts covered in lab - requirements, collaboration, communication, and compromise
- Demonstrate their problem solving ability by installing iTrust on their systems
- Understand their expectations and responsibilities in CSC 326 (the TA is not going to hold their hands - they are expected to show initiative by completing pre-labs and solving problems on their own)

Schedule

- Introductions (15 minutes)
- The Drawing Lab Activity (60 minutes)
- Install iTrust (45 minutes)
- Discuss HW1 (10 minutes)

Strategies and Procedures

Focusing activity to begin lesson:

We'll begin with introductions. I'll have each student begin by telling us their name and what they wish to do with their degree in Computer Science when they graduate. I want to make this class a forward-thinking group of students who take what they learn personally with their graduation goals in mind.

Instructional Strategies: What are you/your students going to do for each learning outcome?

The activities are predefined.

The introduction activity, because it focuses on their graduation goals, will help set the stage for students to take the course seriously. This course is not about programming, and they must see this immediately. I can talk about my research if the point isn't made clear or if they don't take the questions seriously.

The drawing activity will convey that there are two ways to look at requirements: as a minimum goal to achieve or as a challenge to be beaten. Some students will see the requirements as constraints, exhibiting creative approaches to the problem. The themes that will be covered are "requirements", "collaboration", "compromise", and "communication".

I expect the students to be capable of following instructions and seeking help when they need it. They will first be asked to install iTrust on the lab machines together, and then - if they have time - be free to install it on their own machines after they see how it works. The students are free to use their laptops if they like, but getting the software installed is their responsibility. I'll encourage students to help one another install iTrust if they get it done themselves.

Evaluation and Feedback

How do you know if they "got it"? What classroom assessment techniques could you use?

Each of the activities are scaffolded. In the drawing exercise, I'll ask them all why they think we do each activity and what each activity means with guiding questions (provided by the activity handout). Their answers should get more accurate with each iteration, converging on notions that requirements are loose, and they will be required to exercise judgment as they complete the exercises.

The installation exercise is also scaffolded. By requiring students to install iTrust on the lab machines before installing on their laptops, they will work in a simple environment before moving to a more complicated one. I don't expect the students to get done in class, as this class involves a significant amount of work outside of the classroom.

Follow-Up and Conclusion

How are you going to end class? Think about the "So What, Now What?" question.

HW1 is going to be tough. You have a week to learn how to use 6 new technologies. I'm leaving it up to you. As computer science students, I think you're capable of handling that.

Reflection

Before Class/Lab

What materials/resources do I need?

What technology/tools will I need to gather?

iTrust v15 should be on SourceForge.

Do I need to test an experiment or practice any activities ahead of time?

I need to make sure that the right versions are still on the lab machines. They shouldn't have changed since Fall, but I need to make sure that iTrust can still be installed.

What other special considerations should I prepare for?

After Class/Lab

How did I deviate from the lesson plan?

I was not as hard on the students as I planned to be. I must not have it in me to be the drill-sergeant type of instructor. For the most part, everything went as planned, but that's because it was heavily scripted.

What worked well in this lesson?

I felt like we stimulated some good discussion by talking about the reason why students are in CSC.

Requiring the students to do the work on the lab machines before their laptops was an excellent way to scaffold the assignment. They were able to solve the installation of iTrust mostly independently, so this should help prop them up for success when it comes to installing it on their laptops and doing HW2.1.

What didn't work well?

There was a bug in the presentation animation where the "secret" to the collaboration exercises was displayed with the instructions, kind of ruining how the exercises were

supposed to work. I tried to connect the concepts of requirements and teamwork in spite of that, but it was difficult.

What should I continue to do when I teach?

I liked the food for thought approach for starting off class. I think what I'm going to do is before each lecture, post a brief snippet of a faculty member's research and encourage the students to think about how they would respond to the challenges that they face.

What might I do differently next time?

Next lab, I'm going to focus substantially on proper test case and data development. Students, far too often, don't seem to grasp the importance of testing, so I may give them a pop quiz if Dr. Xie doesn't beat me to it in the lecture segment. Last semester, I feel like I didn't cover regression tests early enough to make a difference... we'll see if we can fix that this semester.

Notes

I would like to thank Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the LaTeXSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.