

# CSC 326: Software Engineering

## Lab 3 Lesson Plan

Barry Peddycord III

30 Aug 2012

### Overview and Purpose

#### Lesson Topic

Writing fault reports and project estimation.

#### Learning Outcomes

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

- Identify faults and reproduce failures in iTrust (Knowledge)
- Create a fault report (Application) - a fault report should include objective details, such as how to reproduce the failure, not “low-level” details, such as coding suggestions (Eric S. Raymond’s reporting good bugs)
- Navigate iTrust (Experience)
- Explain the rationale behind test-driven development (Understand)
- Estimate workload commitments associated with fixing faults (Evaluate)

### Strategies and Procedures

#### Focusing activity to begin lesson:

We’ll begin by walking through the homework assignment. Show students the bugzilla interface, explain the expectations, and then let them loose.

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

- Lab exercise: students will reproduce errors and record faults in bugzilla.

- We'll play planning poker to get a feel for how estimating time commitments will be.
- Finally, wrap up with a discussion of test-driven development and why accurate estimates are important - namely that students will not be "coding" when they get out of college, but will assume leadership/management positions.

## **Evaluation and Feedback**

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

For the first part, I will be walking about the room while students reproduce the errors and fill out the reports on bugzilla. I'll ask them questions on the spot to gauge whether they get it.

During the estimation portion of the lab, I'll give think-pair-share a shot. Have students (with their partners), tell me why estimation is important.

## **Follow-Up and Conclusion**

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

I'll probably end this by telling them a compelling story. This is where I'll make my statement that project estimation is the most error-prone part of software development, but it is quite arguably the most important.

## **Reflection**

### **Before Class/Lab**

#### **What materials/resources do I need?**

Planning poker cards, Slides

#### **What technology/tools will I need to gather?**

Bugzilla, Overhead Projector, Google Docs Slides

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

I should be able to reproduce the errors that the students need to locate

**What other special considerations should I prepare for?**

## **After Class/Lab**

**What worked well in this lesson?**

Having the focusing activities in order to turn the lab into a captive audience was very effective. I'm able to corral the students into doing one thing and have them pay attention to me.

**What didn't work well?**

Trying to give a "final speech" doesn't work particularly well since folks are always working up until the bell.

**What should I continue to do when I teach?**

I should break the lab up into definitive focusing activities. I don't need to talk on and on. Just get everyone quiet and explain what's happening for today's assignments.

**What might I do differently next time?**

For the final "focusing" activity, give everyone an index card and ask them to write down a single question they have about the course material. Also, interleave more questions into my talks to make sure they are thinking and paying attention. It's not good to talk forever, but it's just as bad to only talk for five minutes.

## **Notes**

Thanks to Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.

# CSC 326: Software Engineering

## Lab 4 Lesson Plan

Barry Peddycord III

6 September 2012

### **Overview and Purpose**

#### **Lesson Topic**

Requirements Inspection, CRC Cards, Project Estimation

#### **Learning Outcomes**

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

- Write CRC Cards for a Java Program (Apply)
- Inspect requirements and ensure that they are feasible and complete (Evaluate)
- Estimate workload commitments (Evaluate)

### **Strategies and Procedures**

#### **Focusing activity to begin lesson:**

Begin the lab by meeting your new partner. We'll do what we did in STARS - each person will tell the other three things about themselves and introduce us to their new labmate.

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

- Students will inspect requirements and fill out an inspection document
- Students will then create and act out scenarios for these requirements using CRC cards
- Students will play planning poker again and come up with estimates on how long implementing these requirements will take.

## **Evaluation and Feedback**

### **How do you know if they “got it”? What classroom assessment techniques could you use?**

I will follow the strategy that seems to work the best: introduce activity, explain its meaning, have students perform it, bring everyone back together, ask questions about the activity. By asking these questions, I can figure out if they see how these activities are tying in to the big picture.

- Who remembers what CRC means? (Class-Responsibility-Collaboration)

## **Follow-Up and Conclusion**

### **How are you going to end class? Think about the “So What, Now What?” question.**

The last “activity” will be to write down a “complaint card” where students will write down one problem they have with the lab. This will not only give me feedback on my teaching, but will also focus them on something at the last minute.

## **Reflection**

### **Before Class/Lab**

#### **What materials/resources do I need?**

Index cards, Planning poker cards, Slides

#### **What technology/tools will I need to gather?**

None!

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

I should practice with CRC cards for the proposed requirements.

#### **What other special considerations should I prepare for?**

### **After Class/Lab**

#### **What worked well in this lesson?**

Using guiding examples to steer the class in the direction I wanted to go worked really well. I was able to get the material across, even though I only had about an hour to prepare.

**What didn't work well?**

Setting up the assignment the night before. Everything that went wrong today had to do with a lack of planning.

**What should I continue to do when I teach?**

Continue to have “focus activities” to get everyone’s eyes back on me. I don’t need to be so timid about breaking up student discussions during the lab session.

**What might I do differently next time?**

Try a different “ticket out the door”. Maybe a clearest point/muddiest point.

**Complaint Cards**

The two big complaints were the lack of ability to choose partners (which they get to do for the last project) and the lack of clarity. The earlier class believed that the requirements were simply not clear, so I’m going to work on being more clear in advance.

**Notes**

Thanks to Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.

# CSC 326: Software Engineering

## Lab 5 Lesson Plan

Barry Peddycord III

13 September 2012

### Overview and Purpose

#### Lesson Topic

Peer-review of another team's design and test plans.

#### Learning Outcomes

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

- Evaluate and provide feedback on a design plan (Evaluate)
- Evaluate and provide feedback on a test plan (Evaluate)

### Strategies and Procedures

#### Focusing activity to begin lesson:

Break students into pairs of pairs.

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

- Students will peer review each others' designs and test plans

### Evaluation and Feedback

#### How do you know if they “got it”? What classroom assessment techniques could you use?

At the end of each segment, a pair will describe their peer's design/test plan and then tell us the feedback they gave on it. If I see errors, I'll get the class to participate in helping me correct them.

## **Follow-Up and Conclusion**

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

N/A - I'm going to let students work on HW3 this time.

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

Today was a “get your homework done” day. Getting the students to present what they learned from the peer review was a great way to assess them.

Furthermore, by simply requiring students to get their technology working before leaving, I'm able to make sure that technology concerns do not pose a problem for the next homework as well as dealing with potential “it wasn't working” issues down the road. When I make it a requirement in lab to have the technology running, there's no excuse later on.

**What didn't work well?**

The presentations were shorter than I would have liked, and the students got ahead of the material meaning that between presentation one and presentation two, they didn't actually have to do the work. I'm sure some were goofing off, but as I mentioned, I was troubleshooting.

**What should I continue to do when I teach?**

Student presentations are a great way to put students on the spot, helping them realize holes in their understanding.

**What might I do differently next time?**

I really winged the presentations. Before students give a presentation - even an informal one - I should come up with a quick rubric to evaluate them based on what they tell us so that I make sure they tell us everything.



## Notes

Thanks to Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.

# CSC 326: Software Engineering

## Lab 6 Lesson Plan

Barry Peddycord III

20 September 2012

### Overview and Purpose

#### Lesson Topic

HW4 preparation - just like HW3, but solo. Since students have been doing this all semester, it's time to work in preparation for the midterm.

#### Learning Outcomes

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

- Perform requirements inspections (Analyze)
- Plan out the scenarios using a sequence diagram (Apply)
- Design the project using UML (Apply)

### Strategies and Procedures

#### Focusing activity to begin lesson:

Begin talking about the midterm and performance on HW3. Not fun, but needs to be done.

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

Today will be mostly freeform planning. I'll want a concrete plan of action from everyone before they leave, but they can't work on the artifacts together. Before we leave, we'll come together for 15 minutes or so to talk about what everyone is thinking of doing.

## **Evaluation and Feedback**

### **How do you know if they “got it”? What classroom assessment techniques could you use?**

I'll sit with each group for 10 minutes or so and join their discussion. I'll have students turn in a UML Diagram before they leave.

## **Follow-Up and Conclusion**

### **How are you going to end class? Think about the “So What, Now What?” question.**

Let's end by talking as a class about things that need to be changed. “Ticket out the door” is a UML diagram.

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

I need to come up with a reference UML and Sequence Diagram for this project so that I can guide students in the right way.

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

The class delivery was extremely smooth. We started and ended on time and fleshed out the requirements more deeply than we did for HW3-1.

**What didn't work well?**

Somehow, I ended up making a mess and not knowing that the functionality for HW4 was already present in iTrust. Luckily, I was able to bounce back and flesh out a new project.

**What should I continue to do when I teach?**

More large group discussions. These seem to work really well since group discussions are kind of my thing.

**What might I do differently next time?****Notes**

Thanks to Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.

# CSC 326: Software Engineering

## Lab 7 Lesson Plan

Barry Peddycord III

27 September 2012

### Overview and Purpose

#### Lesson Topic

HW4 peer review and static analysis.

#### Learning Outcomes

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

- Analyze a design and test plan (Analyze)
- Use static analysis tools to find bugs (Apply)

### Strategies and Procedures

#### Focusing activity to begin lesson:

Talk about static analysis - what it is and how it works.

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

Like last time, students are going to inspect one another's designs, test plans, and test data. They will also work with static analysis tools in Eclipse to find bugs in their code.

### Evaluation and Feedback

#### How do you know if they “got it”? What classroom assessment techniques could you use?

Unlike last time, I'm going to try something new: students are going to be given the rubric and asked to evaluate each other's project reports with particular emphasis on

the UML diagrams and test plans (they won't be asked to look at the test data or implementation). I'll make a new Google form specifically for this purpose.

## **Follow-Up and Conclusion**

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

Talk about static analysis - the research that our department does.

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

Make a special peer-review rubric specifically for HW4-1.

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

I should play with the FindBugs tool and make sure I know how it works.

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

The more directed peer review google form (where I gave the students the rubric) worked substantially better than giving them a generic form to fill out. I wish I could have started them off this way, so that their presentations would have been a bit deeper.

**What didn't work well?**

**What should I continue to do when I teach?**

Give students more guidance on what matters when conducting peer review.

**What might I do differently next time?**

## **Notes**

Thanks to Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.

# CSC 326: Software Engineering

## Lab 9 Lesson Plan

Barry Peddycord III

11 October 2012

### **Overview and Purpose**

#### **Lesson Topic**

Final Project: Requirements Inspection

#### **Learning Outcomes**

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

- Do a requirements inspection.

### **Strategies and Procedures**

#### **Focusing activity to begin lesson:**

We'll just get right down to business. I want to give the students plenty of time to work.

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

Do the exercise and hit the ground running.

## **Evaluation and Feedback**

**How do you know if they “got it”? What classroom assessment techniques could you use?**

## **Follow-Up and Conclusion**

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

I’m going to spend five minutes at the end of class giving back midterms.

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

I need to get the students their midterms.

**What technology/tools will I need to gather?**

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

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

**What didn’t work well?**

Somehow or another, the requirements for this lab didn’t get conveyed clearly.

**What should I continue to do when I teach?**

**What might I do differently next time?**

I think I need to focus more on going over the instructions together. Perhaps say “read instructions for 5 minutes” and then ask what’s going on. Since this is a lab, not a lecture, it’s absolutely important that everyone is on the same page.

## **Notes**

Thanks to Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.



# CSC 326: Software Engineering

## Lab 10 Lesson Plan

Barry Peddycord III

18 October 2012

### **Overview and Purpose**

#### **Lesson Topic**

Final Project: Iteration 1

#### **Learning Outcomes**

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

- ...

### **Strategies and Procedures**

#### **Focusing activity to begin lesson:**

Explain the changes in the requirements since last week.

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

### **Evaluation and Feedback**

#### **How do you know if they “got it”? What classroom assessment techniques could you use?**

This class is about me going around the room and evaluating the student artifacts and having them do their first coding. I'll be giving each group some very directed feedback so they know what they need to do for the rest of the project.

## **Follow-Up and Conclusion**

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

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

Giving student teams the extended feedback was tremendously rewarding and helpful for them.

**What didn’t work well?**

**What should I continue to do when I teach?**

**What might I do differently next time?**

## **Notes**

I would like to thank Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot] edu*.

# CSC 326: Software Engineering

## Lab 11 Lesson Plan

Barry Peddycord III

25 October 2012

### Overview and Purpose

#### Lesson Topic

Final Project: Iteration 2 and Security (Observed)

#### Learning Outcomes

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

- Understand the causes and consequences of SQL and XSS Attacks
- Perform SQL and XSS Attacks (Apply)

#### Schedule

0:15 for intro and demo

0:25 for students to do security exercise

0:20 for student presentations

0:50 for students to work on iteration 2 (will give feedback for iteration 1)

Total time 1:50

### Strategies and Procedures

#### Focusing activity to begin lesson:

Ask students to recap yesterday's lecture on XSS and SQL Injection (what causes them) while demonstrating each of them on the vulnerable version of iTrust.

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

We'll do the prescribed exercise for lab 11 - each group of students will do two of each attack, and demonstrate one of their attacks on the overhead computer.

## **Evaluation and Feedback**

### **How do you know if they “got it”? What classroom assessment techniques could you use?**

Going to use class demos - each group will present one of their attacks for a total of four demos in section 4 and three in section 2.

## **Follow-Up and Conclusion**

### **How are you going to end class? Think about the “So What, Now What?” question.**

I’m going to use a google form and ask students to fill out how they would fix the security holes they found.

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

Come up with a few easy attacks to use as demos before breaking the students up into teams.

**What other special considerations should I prepare for?**

Dr. Heckman will be observing section 4.

### **After Class/Lab**

**What worked well in this lesson?**

The students were able to find most of the security faults I expected they would be able to find.

**What didn’t work well?**

My demo had a bug in it, so that slowed down getting everyone on the same page. In general, the demo was done too quickly and probably didn’t help anyone in the first place.

**What should I continue to do when I teach?**

My conversations with the students, hearing them articulate what they were thinking and how to solve the problems at hand.

**What might I do differently next time?**

I should really take advantage of “flipping” the classroom. After today’s botched demo, I really think it would have been more helpful to have students practice setting up the lab materials at home before coming to lab. Usually, I’m afraid that students will complete the lab exercise before coming to class, but when that’s the case, it just means they can help the rest of their team and explore the problem in even greater depth than the hour would permit if we all started together. Next time I do a lab like this (which won’t be this semester, since this was the last exercise-based lab), I’ll utilize pre-lab activities more regularly.

**Notes**

I would like to thank Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the  $\LaTeX$ Source, please contact me at *bwpeddy@ncsu.edu*.

I would also like to thank Dr. Sarah Heckman for observing this lab and giving me her thoughtful feedback!

# CSC 326: Software Engineering

## Lab 12 Lesson Plan

Barry Peddycord III

1 November 2012

### **Overview and Purpose**

#### **Lesson Topic**

Final Project: Iteration 3 and Consultation

#### **Learning Outcomes**

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

- Articulate questions for industry experts visiting the lab (Evaluation)

#### **Schedule**

Each team will get 30 minutes with the developers from Premier Health to show their projects and learn a thing or two about testing.

### **Strategies and Procedures**

#### **Focusing activity to begin lesson:**

Introduce our guests from Premier.

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

N/A

## **Evaluation and Feedback**

**How do you know if they “got it”? What classroom assessment techniques could you use?**

N/A

## **Follow-Up and Conclusion**

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

There will be an evaluation regarding the guest visit. I would like to hear the students’ feedback on the guest visit.

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

**What other special considerations should I prepare for?**

We will be having guests, so I need to ensure that they understand how prepared our students are.

### **After Class/Lab**

**What worked well in this lesson?**

The guest speakers were very engaged, but I think they got fatigued by the end of the day.

**What didn’t work well?**

**What should I continue to do when I teach?**

**What might I do differently next time?**

## **Notes**

I would like to thank Dr. Barbi Honeycutt for the great lesson plan template! For anyone who wants the L<sup>A</sup>T<sub>E</sub>XSource, please contact me at *bwpeddyc [at] ncsu [dot]*

*edu.*



# CSC 326: Software Engineering

## Lab 13 Lesson Plan

Barry Peddycord III

8 November 2012

### **Overview and Purpose**

#### **Lesson Topic**

Final Project: Final Iteration

#### **Learning Outcomes**

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

- Finish their project

#### **Schedule**

It's a working day. Towards the end of the lecture, however, I want to poll the class for ideas for next semester to give the next TA something to work with.

### **Strategies and Procedures**

#### **Focusing activity to begin lesson:**

Ask how they liked the visitors.

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

N/A

## **Evaluation and Feedback**

**How do you know if they “got it”? What classroom assessment techniques could you use?**

N/A

## **Follow-Up and Conclusion**

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

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

**What didn’t work well?**

**What should I continue to do when I teach?**

**What might I do differently next time?**

## **Notes**

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

# CSC 326: Software Engineering

## Lab 14 Lesson Plan

Barry Peddycord III

15 November 2012

### **Overview and Purpose**

#### **Lesson Topic**

Final Project: Final Presentations

#### **Learning Outcomes**

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

- Present their work
- Answer questions about their work

#### **Schedule**

Each group will give a presentation (about 10 minutes each) The class will vote on presentations (secretly) Groups will pair up and peer-evaluate each others' presentations

### **Strategies and Procedures**

#### **Focusing activity to begin lesson:**

N/A

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

The students are going to break into focus groups to help identify problems in their presentations. I want them to be able to judge each other on content and form.

## **Evaluation and Feedback**

### **How do you know if they “got it”? What classroom assessment techniques could you use?**

The peer evaluations. I want each group to try to figure out if they were able to win the competition. I'll announce the winner at the end.

## **Follow-Up and Conclusion**

### **How are you going to end class? Think about the “So What, Now What?” question.**

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

I need to prepare a rubric for the voting.

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

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

The students had some really excellent conversations when peer reviewing each others presentations. The internet was down today, so we could not do the google forms rubrics, but the unstructured discussion went better than any technology would have done.

**What didn't work well?**

I had to make a hard call between two excellent teams today. Having to justify the grains of sand that tipped the scale was difficult, and it felt like I hurt some feelings with that call.

**What should I continue to do when I teach?**

I wish unstructured organic discussions would occur more often. It was interesting how well the students were able to have a fruitful discussion when the other TA and I *left the room* to deliberate on the winners.

**What might I do differently next time?**

## **Notes**

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

# CSC 326: Software Engineering

## Lab 15 Lesson Plan

Barry Peddycord III

29 November 2012

### Overview and Purpose

#### Lesson Topic

Reflection

#### Learning Outcomes

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

- Reflect on their learning over this semester
- Apply this knowledge to solving an old homework assignment
- Mentally prepare for senior design

#### Schedule

Ask students to look at HW3 and reflect - as a team - on how they would solve it. Have students discuss and then bring the class together.

I'll try to put things in perspective to drive the discussion to be around senior design, as many students will be taking that next semester.

### Strategies and Procedures

#### Focusing activity to begin lesson:

"Assign" HW3.

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

I'll be assigning HW3 in order to make them think about the old project in a new light, now that they know iTrust.

## **Evaluation and Feedback**

### **How do you know if they “got it”? What classroom assessment techniques could you use?**

I'll facilitate a focus group to have students reflect on what they've learned and how they would tackle a problem given their experiences from this semester.

## **Follow-Up and Conclusion**

### **How are you going to end class? Think about the “So What, Now What?” question.**

Don't cry... ;-;

## **Reflection**

### **Before Class/Lab**

**What materials/resources do I need?**

**What technology/tools will I need to gather?**

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

**What other special considerations should I prepare for?**

### **After Class/Lab**

**What worked well in this lesson?**

The focus group was excellent - we were able to have a great discussion about what the students really wanted to see in the class and what they feel as though they got out of it. They followed the exercise pretty well, and it turned out to be a great way to have students think critically on what they wish they would have done had they had the time.

**What didn't work well?**

Many of the students were done with their projects already, so I didn't try to keep them if they left early.

**What should I continue to do when I teach?**

**What might I do differently next time?**

## **Notes**

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