

Arthur Liou  
CS362

### Extra Credit – Course Feedback

I am going to have the opportunity to help rewrite CS362 over the next couple of terms and I would really like to hear your thoughts on the current course. This survey is worth 50 extra credit points.

1. What is your general opinion of the content in CS362?

I enjoyed the content in CS362 a lot. A great class and thought it was a great learning tool / requirement for any CS student. However, instead of calling the class Software Engineering 2, I think the class should be renamed to include “Testing”, like Software Engineering 2 – Testing or Software Engineering 2 – Software Testing

2. What do you want to see more of?

I thought this was a great mix of classwork / teamwork, real world projects, solid midterm/final ratio, and the questions asked.

3. What was helpful?

Yes and no. I currently work as a software engineer, so I touch upon many of the testing concepts in the course (particularly the first half) and didn’t learn much new there. But yes because I do not test in C or Java, so learning how to test and debug in those were great learning experiences.

4. What wasn’t?

See above :). But I also thought only having resources to test in Eclipse and IntelliJ were outdated and forces students to use tools that may / may not be preferable in the work place. For example, at my startup, part of our codebase is in Java and an

5. Are there any topics that should be given additional coverage?

N/A. I think this format is great as is.

6. If so, what material should be given less coverage to make room?

N/A

7. Should there be more content in each week?

I think this was a manageable amount for a 4 unit class. Keeping in mind that some students in this class work full time, work part-time, could be an athlete, etc., I found that this course shared a great number of material and concepts that we could *learn* and not just blindly regurgitate.

8. Do you think using Dominion as a code base worked for you?

Yes, I found this to be a fun exercise, in Refactoring/Testing/Introducing new bugs, and also learning a new card game!

9. Do you think using UrlValidator as a code base worked for you?

Absolutely not. On the most positive note regarding the assignment, Part B was a fiasco at the very best. Part A was great in helping us understand the code base.

10. Would you prefer to do testing using other languages besides C and Java?

Personally, I do, because the code I work with on a daily basis is not in C nor Java. Working with C was great for OSU students with little to no outside coding experience because most of our previous courses are in C/C++. But working in Java provided to be harder for most to grasp quickly for UrlValidator.