Course Survey CPRE 556 Scalable Software Engineering Spring 2010

We would like you to assess the course by answering a few questions. We are interested in understanding your perception of the material taught, the way it was taught, and the overall value you got out of the course. We would like to get your response by Tuesday, April 27.

1. Did the course succeed in teaching material that you think will be valuable to you ? What material did you find particularly useful?
2. Was the course taught effectively? What did you like (or not like)?
3. Were you challenged to think? Did you enjoy the challenges?
4. What are the ways in which you think the course should improve?
5. Was the course different than what you expected? Do you have any suggestions for a better description of the course for future students?

See the following pages for student responses to the above questions.

Open Text Responses:

1. Yes, the course is valuable to me. I found it very useful to learn about the principles of good software design.

2. Yes, it was. I liked the way the professor interacted with students and sometimes told anecdotes in physics.

3. Yes, I was. I especially liked the challenge problem about memory leak, even though it was not graded. The word "challenge" really provoked my desire to solve it.

4. It would be interesting to have more real-world examples of large software that make students understand more about the difficulties that big companies encounter.

5. Yes, it was somewhat different. The course was to analyze some characteristics in large software, not to learn how to make software scalable as I had expected. This point should be made clearer in the description of the course.

1. I think this course taught me a lot of valuable material. I learned a lot more about the C programming language (I hadn't used it a lot before) and also a lot about analyzing code. I had not done a lot of hands-on analysis of code before and think it is a very useful skill to have.
2. I think the course was taught fairly effectively. I appreciated that there was a later due date on our assignments for off-campus students, being one of them. This made it much easier for me to complete my assignments on time.
3. I was definitely challenged to think. I believe it was good for me and I learned a lot from it. Sometimes, I did not enjoy it because I felt as though I didn't have enough previous knowledge to reason effectively about the subject, but overall, I appreciated the challenging thinking required.
4. It was sometimes hard for me to keep up with the deadlines for asking questions on projects. I understand the need for them, but was sometimes not able to ask any questions because I did not have time to start the project early enough.
5. The course was basically what I expected although I did not expect to only do analysis of code, and no actual writing of code. I very much liked that format though, and wouldn't have changed it.

Thanks!

1. The course has done a very good job in teaching material relevant to my area of interest. The projects really took a lot of work and made me struggle, but were great learning aids.

2. The course was taught very effectively. I specially liked Dr.Kothari teaching methodology where simply reading and believing is not enough. One needs to prong deeper and question what has been printed and why. I have become a fan of his recurrent quote which talks about reading from the masters of any subject if one needs to learn about a subject. I have followed his tips and tricks in other classes and they have deeply benefited me.

3. The course not only challenged me to think, it also taught me how to dwell over subject matter.

4. I can't think of any way this course can be made better.

5. The course was better than what I expected.

I really liked the course CprE 556.

Here are the reasons:-

1. It helped in true learning process, as there was no fear of any exam.
2. The evaluation component which are 3 projects in this course, covered every concepts covered in the class, also as the size of the designed project was efficient in a way that it will help further in future in most of the software engineering problems.
3. The homeworks helped in learning of the software engineering concepts which are useful in day to day life.
4. The project which we did in as a part of the course CprE 556 are really useful as they cover most of the software engineering practical problems.

I really liked the way prof. Kothari takes the class. It helps to broaden the knowledge area. Even after the class, the thought process continues and we try to look into the problem from different angles.

Yes, the class makes us thinking about the challenges. I was really challenged by the project -2.

The course was different than I expected because as the name suggest scalable software engineering I thought that it will be advance of what we leaned in software engineering course but the class was really interesting from the day it started till today as it covered the practical problems and we learned a standard method to solve these problem.

Thus the course was really very interesting and i was challenged to think more and more and I would like to major in Software Engineering.

Open Text Responses:

Overall assessment: I learned a lot about interesting topics on sw engineering through the course. Many thought provoking topics covered. Instructor's teaching methodology very clear and descriptive on learning material, instructor paid attention to course material relation to industry practices.

Other comments:
1. Some questions in HW 1-3 were not covered in class, basically students had to get educated through google search; maybe some practical scenarios or problem solving regarding those questionnaire in class would be more related.
2. The guideline that points will be taken off for late submission should have been mentioned in the syllabus to avoid misleading information.
3. It was good that projects were posted and given a lot of time to think about, but I found it difficult to relate to the related lecture material since other new topics were covered between the related lecture for the project and the submission of the project. Especially for off-campus students who have very busy schedule.
4. The other point that bothered me is that project solution was discussed but directions (since the solution can be very abstract and can vary on student to student thinking prospect) were almost like intentionally skipped. For example for project 2, I did not think about including parent nodes (I noticed one student in class had the same question) for finding the minimum graph, I do remember the instructor mentioning about tracing type but it was not clear, so I may have missed the point.
5. Setting a deadline to ask a question about an assignment a week or more earlier is not always helpful. I understand the instructor wanted to be strict but if he wants to instigate learning then setting a deadline weeks before submission of an assignment may not be helpful. That student may not have started the assignment because of other priorities.

- Did the course succeed in teaching material that you think will be valuable to you? What material did you find particularly useful?

Actually the courses succeed in teaching the proper material that I was looking for. Before I registered for the course, I was hesitating as I looked through different related courses at other universities, and they were talking theory! And the software engineering is not a theory!

I liked the flow of the course, where we started with general aspects, concepts, critical issues in software development, and the state-of-the-art software engineering. Then, we moved to study in brief multiple books, and different topics of critical and open issues that one can take as research problems.

For me, I liked the matching pair property, the debugging material, and model-based development more than others, maybe because I was so interested and eager to learn more about such challenging problems.

I took a software engineering course in bachelor, but I didn't get the idea of having a science like that. I was just asking myself, why people would develop or come up with science that cannot be useful to anyone. Now, after I took this course, I figured out that, we really need more deep studies and concentration on this field, as it’s the major place, where we can then succeed in building well-designed software.

****

- Was the course taught effectively? What did you like (or not like)?

I believe that it was taught in an excellent way, Thanks for you Dr. Suraj. This course enriches my information.

I liked the way you mix between real stories and theories, and also the way you adopted the psychology slides into software related topic.

I liked the way, we looked at the problems, and how we discussed the problems in each of the books that we studied and also the papers that we looked at.

Going through the papers with you was interesting, as I was looking how such a professional academic person having great experience look at the papers and how he analyze them, and I really learned a lot, and I am using such techniques while reading papers.

I liked the pro

Open Text Responses:

jects, and the homeworks so much, as they were entertaining and enriching my information, through thinking and looking at different resources for information. It also enhanced my way of thinking and the ability of good summarization and communication.

I liked the flow of projects, as each one of them needs the previous experience, and we have gradually moved from simple one on XINU to LINUX based one.

****

- Were you challenged to think? Did you enjoy the challenges?

Yes of course, this is what I liked the most! Especially on the homeworks and projects. Also on the discussions that were following each lecture set.

Sometimes, while I am trying to solve a homework problem or a project, I was eager to solve it and write a good answer.

****

- What are the ways in which you think the course should improve?

Go through more papers, not in class. But, at least the students should go through some assigned papers and maybe summarize them and address the open issues and other related info. For example, we can take a paper describing some software metric and try to criticize that metric!

At the end of semester, we can go over the first topics that we discussed, and take the projects and the experience we get through the course, and try to identify whether the issues that we discussed are valid or not!

****

- Was the course different than what you expected? Do you have any suggestions for a better description of the course for future students?

Yes a lot. I don't have an alternative description to what is in the syllabus. Its good, but the way the course is taught is totally different and it’s really challenging and entertaining!

****

In general, I liked this course!

Thanks a lot Dr. Suraj, your efforts are highly appreciated! I liked every tiny detail in this course, as I sometimes watch the lecture one more time at home to get all the details noted in my notes!

Open Text Responses:
The course provided new insight into evaluating software for defects. The 4 basic concepts of comprehension complexity, code decay, ripple effect, and estimation unpredictability were not new. But the explanations in class made me look at them from a different angle. 2. I like the way the course was taught where each topic built on the previous topic. I felt the course was well organized and Professor Kothari used the time effectively. The in class demonstrations of how to analyze large software using the atlas tool and how the analysis related to the class topics was most effective. One thing that would have been helpful was to have had the lecture slide handouts available in advance so I could review the topics prior to watching the lectures. 3. The lectures, homework, and projects all provided a challenge, The in-class discussions and questions were effective at helping me to comprehend the topics. The paradigm of trying to analyze code in different ways was quite challenging, and very enjoyable. The great thing about a Kothari class is that he always makes you think. 4. I would recommend adding a homework assignment to write a program in C that uses pointers and global variable assignments as a primer for help with the code analysis of xinu and linux. 5. I thought we might cover some topics related to the implementation scalable software across clusters or multiple servers. The class explored several topics, one think that I think would be helpful is to set up a class blog where students could comment on lectures and the papers that were discussed in class. In some cases it seemed optional to read the papers, Maybe have a list of 7 - 10 papers on software engineering and then require us to read 4 of them and blog on our thoughts. This was something we did in another class and allowed me to freely express my thoughts on class topics and reading assignments.

Thoughts on CprE 556...

Sent: Sunday, March 28, 2010 12:15 AM

To: Kothari, Suraj C [E CPE] [kothari@iastate.edu]

I would like to express my surprise and excitement in regard to CprE 556. This is my first graduate level class; I graduated from Louisiana State University in 2001 with a Bachelors of Science in Computer Science and a minor in Mathematics (Calculus and discrete). Initially, I expected to have to dust off my Calculus book and relearn the semantics of Greek symbols and obscure theorems. However, I have found your class to be full of very relevant and modern ideas about software engineering. I did not realize that university professors are thinking about the problems that I face daily while working "in the field" and far away from the class room. Some of the topics that I found particularly motivating:

1. Program slicing
2. Analysis of matching pair defects
3. Model-based programming and testing
4. Assistance is how to critically read academic papers
5. Analysis of the Linux kernel (yet to be done as of this writing)

I hope this is not offending, but I thought software engineering at the graduate level was about useless software metrics, business processes and more of the same ol' hot-technology-of-the-moment propaganda. Note, I believe such things are important, but I can easily absorb such information on my own, either through my employer's training or books from Barnes n' Noble. I appreciate that you focus on fundamentals, whether "old" or "new", and show how new technology can often break down into such fundamentals.

I feel qualified to stamp my approval and recommend this course to my peers. I currently have eight years of industrial safety/mission critical software engineering experience. I have worked for Lockheed Martin (Joint Strike Fighter), Northrop Grumman (Joint Strike Fighter), EDS (Travelocity server system). I currently work at Rockwell Collins where I helped to develop a Terrain Awareness/Warning System for commercial aircraft and software-based military radios. I have seen a wide spectrum of software quality (both in terms of processes and products) and feel passionate about making a difference in my industry.

Dr. Kothari, I look forward to meeting you on Thursday and discussing my academic future. I hope to interlace my academic career with my industrial career and would like any thoughts you have on how to most effectively do that.

Regards,