

*** PROVISIONAL REPORT ***

UNIVERSITY OF TEXAS AT AUSTIN
 Downing, Glenn P C S373 51045
 E100 EXPANDED

COURSE-INSTRUCTOR SURVEY
 SOFTWARE ENGINEERING

Spring 2019 DEPARTMENT COPY
 Grade-eligible enrollment = 55
 Surveys Returned = 51

	NUMBER CHOOSING EACH RESPONSE					NO. REPLIES THIS ITEM	AVG.
	Str Disag	Disagree	Neutral	Agree	Str Agree		
1 COURSE OBJECTIVES DEFINED-EXPLAINED	0	0	4	19	28	51	4.5
2 INSTRUCTOR PREPARED	0	0	0	9	42	51	4.8
3 COMMUNICATED INFORMATION EFFECTIVELY	0	0	3	15	33	51	4.6
4 STUDENTS ENCOURAGED-ACTIVE ROLE	0	0	1	8	42	51	4.8
5 INSTRUCTOR AVAILABILITY	0	1	2	14	34	51	4.6
6 COURSE WELL-ORGANIZED	0	1	6	14	30	51	4.4
7 STUDENT FREEDOM OF EXPRESSION	1	1	6	18	24	50	4.3
8 CLASS PARTICIPATION ENCOURAGED	0	1	0	13	37	51	4.7
9 ENGAGING INSTRUCTION	0	1	5	17	28	51	4.4
10 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT	0	0	1	12	38	51	4.7
11 INSTRUCTOR EXPLANATIONS CLEAR	0	0	0	18	33	51	4.6
12 GENUINELY INTERESTED IN TEACHING COURSE	0	0	1	9	41	51	4.8
13 HELPFUL COURSE MATERIALS	0	4	9	18	20	51	4.1
14 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS	0	3	17	18	13	51	3.8
15 ASSIGNMENTS AND TESTS RETURNED PROMPTLY	0	1	4	21	25	51	4.4
16 ASSIGNMENTS USUALLY WORTHWHILE	1	0	4	11	34	50	4.5
17 STUDENT PERFORMANCE EVALUATED FAIRLY	0	4	7	14	26	51	4.2
18 STUDENT PERCEPTION OF AMOUNT LEARNED	0	1	2	11	36	50	4.6
	Vry Unsat	Unsat	Satisfact	Very Good	Excellent		
19 OVERALL INSTRUCTOR RATING	0	0	6	13	32	51	4.5
20 OVERALL COURSE RATING	0	1	8	13	29	51	4.4
	Excessive	High	Right	Light	Insuff		
21 STUDENT RATING OF COURSE WORKLOAD	4	28	17	2	0	51	
	Less 2.00	2.00-2.49	2.50-2.99	3.00-3.49	3.50-4.00		
22 OVERALL UT GRADE POINT AVERAGE	0	2	5	12	32	51	
	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>F</u>		
23 PROBABLE COURSE GRADE	30	12	9	0	0	51	

 For the computation of averages, values were assigned on a 5-point scale so that the most favorable response was assigned a value of 5 and the least favorable response was assigned a value of 1.

COMMENTS:

Total Number of Comments: 29

-
1. I learned so much in this course, especially concerning full stack development. I honestly feel much more comfortable learning new tools and technologies on my own.
-
2. It's unacceptable how the grading criteria for each of the phases of the IDB project is still so vague even despite this class having been around for years. The fact that we only know exactly what the graders expect for certain parts of the project a few days before the project is due or we are given arbitrary deadline extensions after overworking to hit a deadline is inexcusable especially given the fact that this is not the first time that this class has been taught. Also grading is super arbitrary and too slow, feedback isn't returned in time for the next phase of the project. I gained exactly what I expected from this class, an overview of web application development, but the structure under which I gained it should have been better.
-
3. Main complaint is the changing requirements for projects on piazza often the day before a deadline.
-
4. I feel like I learned a lot in this course. I enjoyed working in a team setting on a continuous project as it was similar to a work environment. I also thought that the guest lecturers were really insightful.
-
5. I thought his lectures over programming languages were extremely helping in understanding the fundamentals. Specifically, with Python, I took the 1 hour Python course and left more confused, but Professor Downing really helped me grasp Python and I feel like it is now my strongest language. I do wish that he would give a few more lectures on Javascript as well, just so I could have a better understanding of that for the project. The few things I did not really like about this course is the disconnect with the lecture and project. It was clearly stated but it was still a little frustrating. Also, I felt like the exams were very difficult to get partial credit on, which can be very detrimental.
-
6. I think all of the SWE tools that I was introduced to during this class were very useful. I also enjoyed having different speakers come to class occasionally. My only complain is that project requirements were vague at times and were not clarified until very late into the course of the phase. We were given extra days, but these extra days didn't really matter for our group because we had already rushed to finish everything by the original due date which was pretty stressful.
-
7. Very solid instructor. Straightforward, to the point, organized, effective.
-
8. The disconnect between the class and the project was a bit jarring, but other than that I think the class was pretty cool! I found that the more design-oriented stuff was much more interesting than some of the nitty gritty specifics though.
-
9. I will have prefer you focus more on hashing such as dictionaries because it took me a while to understand it.
-
10. I took OOP previously. I like the change in quizzes, including a guaranteed 1 point and taking the top 30. Thank you!
-
11. I think the most valuable part of this class is the project. The lectures were sometimes interesting/seemed useful in the long run, but other times seemed random. My main pros for this class would be: Professor Downing being an engaging lecturer, the ability to work in a team, and the very useful skills I've learned from the project. My main con is that the requirements for each phase of IDB were not clear. The TAs were constantly having to update requirements on Piazza. This made it really difficult to plan ahead and know what we needed to accomplish because we had to search in multiple places to find info. Also even if we planned ahead we had to cram at the end because of new requirements popping up.
-
12. Overall, the course was great. I noticed a couple of areas that would make the instruction and course a little better in my opinion. For one, I felt that it was pretty hard to ask questions during a lecture if I was confused about an implementation, as Pf. Downing was often engaged either solely with the example or the chosen student. I believe that if he took a few pauses during the lecture to address any concerns it would greatly improve the overall learning for quizzes. Secondly, I felt that the grading rubric for the website was really loose, as the true rubric was spaced across a variety of Piazza chains and the official rubric. This made the requirements very easy to miss. It would be helpful to compile it all as the semester closes.
-
13. The class and professor were great. Although there were problems with knowing what was required for the projects. The requirements would change constantly and that made completing the project early difficult. Also, the professor almost never notices when a student raises their hand to raise a question. For the tools for the projects, I found Docker to be too complicated and unnecessary. Also, one of the TA's was fairly rude when we went to him for help. Other than that, the class was great.
-
14. Regarding asking questions in class, some of my classmates had their questions unrecognized due to cold-calling. Hands were raised and not acknowledged. Regarding the exams, though the material may seem fair, it seems that students in competitive program are more likely to perform well. I've heard comments similar to "this is just like competitive programming" from a student who finished the first exam in 10 minutes. Apparently, taking competitive programming will guarantee you an easy 'A' in the first exam
-
15. Lectures: Great! I really like having HackerRank exercises. I noticed a lot of times people would raise their hands in class, but you'd never see them. The refactoring lectures were a bit dry, but I can see the importance of that information. Tools: I'm so happy that we don't use Travis CI anymore. Non-Group Project: I like that we are required to do this on our own. Group Project: I'm not sure who is in charge of this, but the requirements for the projects were not specified until very late into each phase which was very frustrating. Speakers: Opportunities to hear from graduates who are in the industry is great. Tests: Partial credit would be nice for tests we don't pass.
-
16. Lectures: I thought the Python lectures were excellent, although I personally did not gain too much from them as they overlapped heavily with a Python course I had previously taken. The rest of the course felt disjointed at times. Tools: While many of these tools are extremely useful, I felt that this portion of the course became extremely tedious. Collatz: A lot of effort for a relatively contrived project, but otherwise no complaints. Group Projects: These were a good learning experience overall. Speakers: I enjoyed this part of this class. Really unique and something I hope continues into future classes!
-
17. Overall, a really good class. I learned a lot not only through the lectures but also in the projects. The guest speakers were also interesting.
-
18. I don't think the projects were evaluated fairly. It seemed like putting effort into the aesthetics and UI of the site weren't considered as part of the grade, despite the professor and TA's claiming it would. Certain parts of the project were unfairly weighted like commenting on user stories.
-

19. Downing's lecture style is very good and engaging, although I must say that I found the OOP lectures last year more interesting than the lectures for SWE. I think the refactoring lectures in particular were a bit drawn out. I liked the IDB project. It was fun and I learned a lot about React. I used to hate React, but now that I understand it better, I actually like it. So that is definitely a plus for the class. I like the exams on HackerRank. Keep that format for sure.

20. I feel like the grade cutoffs are a little high considering the amount of work and difficulty of the course. Overall though, I learned a lot and enjoyed it.

21. The first exam seemed fairly concentrated and had much material available to study that would accurately reflect the exam. The second exam seems too broad and expects proficiency in many areas that a student may not know. I knew python, regex, and SQL proficiency before the course but I can't imagine being a student learning all this to the degree required. A more clear description of what the exam will contain would help a lot. Also I think going over the basics of software architecture for a few days would help a lot. Not how the tools work technically but where they are used. Such as explaining how docker is used and how the different servers could communicate.

22. Towards the end, refactoring could've been explained more and not as rushed. In addition to that, the second half of the course had fewer class notes and so when I wanted to go back to review it was hard to follow if I didn't really understand the material in class.

23. The lectures are a bit boring, which I suppose is the reason that attendance is required, otherwise no one would show up after the first month. I understand that the professor is attempting to simulate the so-called "real world" with the disconnect between lectures and projects, however I don't think he quite succeeds in this endeavor. His lectures cover basic knowledge of Python and Java, but I would much rather see more in depth exploration of software engineering. Covering refactoring has been a nice excursion and I wish more of the class was devoted to similar topics. The IDB project was fun and very helpful to learn popular web development tools, however I did not appreciate that some of the requirements weren't well specified.

24. I think the biggest gripe I had with the class was in regards to getting project requirements out in time. A lot of the times, we would get specific information regarding Project requirements through piazza posts just days before the project was actually due. It would have been much more helpful if we got this information at least a week in advance and if this information was reflected on the course website as well. Another issue I had with the course was in regards to the blog posts. I think myself and a lot of the other students failed to see the value of doing these weekly posts. If it were up to me I would remove this aspect of the course as it didn't contribute much to my overall experience in the class.

25. This was a great class and I enjoyed it a lot. Thanks Prof. Downing for the hard work you put in to this class. I can tell that you care a great deal about your students. One thing I want to mention is that the group projects is kind of a mixed experience. The reason is that in every group 1 or 2 people carry the whole group. If you already have web development experience (like I had) then the projects are pretty easy. If you don't have this experience the projects are actually really hard. A significant number of students already had webdev experience which led to a large skill gap within groups. I am not sure how to solve this problem.

26. Lecture content was always on-point, very informative and concise, however, with regards to engaging your students: I often felt that you were not giving students enough time to adequately explain their questions and/or responses to yours during lectures, leading to miscommunications, possibly mild embarrassment on their part, and most problematically, a decreased willingness to ask specific questions in class. I don't have much criticism beyond that, this course is top-notch relative to other UTCS courses I've taken; I thought the group project cycle and feedback from TAs and other groups did especially well to simulate a working environment and facilitate good work. Thank you for a wonderful experience! It's been a pleasure.

27. I really enjoyed this class overall. The lectures were really engaging and the HackerRanks that we did in class helped me understand the concepts we were going over (though I wish there had been a RegEx HackerRank in Python). I like using GitLab for the projects and HackerRank for the exams (and I like that the exams are at 6). I wish there were more detailed links to resources like React, AWS, and Mocha though since some didn't have tutorials or had tutorials that didn't work for us. I think the timing for the due date for IDB2 wasn't very good since many group members were gone over Spring Break and I didn't like when the requirements for the projects changed a few days before the due date (google like search for IDB3).

28. I thought that the test did not always demonstrate the ability well. There is only a chance for partial credit if you already get it right. For example, on the first test, I implemented one of the problems almost perfectly but had a syntax error. I didn't end up getting this question correct and lost many points. For the second test, I used the wrong keyword at one point (something that would easily be fixed by Google in the industry) couldn't figure it out and then one of the TA's commented that it might be the wrong word. I instantly remembered the correct word, changed it, submitted well and gained 2 letter grades. 2 letter grades for one word seems not fair.

29. Most of the comments that I want to make about this course I already said about the OOP, but I did feel the projects were a little bit less organized in comparison, both in terms of requirements and in terms of grading. I still am not a huge fan of cold calling, but I do enjoy writing the blog posts.
