*** PROVISIONAL REPORT ***

UNIVERSITY OF TEXAS AT AUSTIN — COURSE-INSTRUCTOR SURVEY — Fall 2016 DEPARTMENT COPY
Downing, Glenn P C S373 51600 — SOFTWARE ENGINEERING — Enrollment = 57
E100 EXPANDED — Surveys Returned = 54

	NUMBER CHOOSING EACH RESPONSE					NO. REPLIES THIS ITEM	AVG.
	Str Disag	Disagree	Neutral	Agree	Str Agree		
1 COURSE OBJECTIVES DEFINED-EXPLAINED	0	1	2	9	42	54	4.7
2 INSTRUCTOR PREPARED	0	0	0	4	50	54	4.9
3 COMMUNICATED INFORMATION EFFECTIVELY	0	0	1	14	39	54	4.7
4 STUDENTS ENCOURAGED-ACTIVE ROLE	0	0	2	5	47	54	4.8
5 INSTRUCTOR AVAILABILITY	0	1	2	12	39	54	4.6
6 COURSE WELL-ORGANIZED	1	1	5	8	39	54	4.5
7 STUDENT FREEDOM OF EXPRESSION	1	1	3	8	41	54	4.6
8 CLASS PARTICIPATION ENCOURAGED	0	0	1	3	50	54	4.9
9 ENGAGING INSTRUCTION	0	1	5	12	36	54	4.5
10 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT	0	0	1	5	48	54	4.9
11 INSTRUCTOR EXPLANATIONS CLEAR	0	1	3	15	35	54	4.6
12 GENUINELY INTERESTED IN TEACHING COURSE	0	0	0	8	46	54	4.9
13 HELPFUL COURSE MATERIALS	2	5	8	13	26	54	4.0
14 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS	0	4	9	11	30	54	4.2
15 ASSIGNMENTS AND TESTS RETURNED PROMPTLY	4	8	10	13	19	54	3.6
16 ASSIGNMENTS USUALLY WORTHWHILE	0	0	5	18	31	54	4.5
17 STUDENT PERFORMANCE EVALUATED FAIRLY	0	3	5	17	29	54	4.3
18 STUDENT PERCEPTION OF AMOUNT LEARNED	0	0	4	20	30	54	4.5
	Vry Unsat	Unsat	Satisfact	Very Good	Excellent		
19 OVERALL INSTRUCTOR RATING	0	0	2	14	38	54	4.7
20 OVERALL COURSE RATING	0	2	7	17	28	54	4.3
	Excessive	High	Right	Light	Insuff		
21 STUDENT RATING OF COURSE WORKLOAD	0	23	29	2	0	54	
	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	0	5	22	27	54	
	__A__	__B__	__C__	__D__	__F		
23 PROBABLE COURSE GRADE	23	22	8	0	1	54	

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. Quizzes are too hard

---

2. I found many of the quizzes felt like trick question because they were asked in a confusing way. The last three projects were difficult, pretty much everything involved in them was new to me and I had to learn everything independently, none of it was covered in class and it didn't use most of the things we did cover in class. I liked to party about refactoring and design I wish that was a bigger part of the class. I never got useful feedback about why I got the grade I did on the first test. I liked vein able type my answers but I think it's important to get the comments and feedback usually written on tests.

---

3. Enjoyed the class!

---

4. Projects and exams were graded very slow

---

5. the tools - git and GitHub are very useful, in both work place and doing projects. Travis was also useful. Collatz and Netflix Collatz was a good way to be introduced to Python and also implementing optimizations we learned in class. Netflix was kind of hacky, and quite dependent on having people get the repos. the group project Slack was very useful for my group. Splitting up the work was beneficial. Some of the tools were a little harder to use, but google was quite useful, as well as otherteam members the speakers Bloomberg, JPL, Pariveda, Spiceworks Quite interesting! Also cool to see other opportunities. quizzes were a bit stressful a big chunk of the grade.

---

6. Easily one of the best lecturers I've had while at UT, I wish you taught some of the lower division core classes. I do wish we were able to have more freedom in choosing the topics of our IDB project though, but it's not a really big deal.

---

7. The entire IDB project seemed great in concept, but was disconnected from what we were learning in class in class we learned quirks in python's design, some SQL, and at the end some refactoring, but the project asked for flask, angular, other web dev tools and project management. I would have preferred not being thrown into the deep end of a bunch of new technology, I can do that myself outside of class. It was joked that we would switch groups part of the way through the project, but I thinkthat would have been better- allowing groups to start fresh on each phase with lessons learned from the previous. Instead poor decisions made early on just worsened as the project continued.

---

8. Tools -got a good understanding of how to use them -should definitely keep using Git and Travis -lab machines don't have PostGres so I couldn't code for IDB3 -never was able to get Docker set up wasted a lot of time on it Group Projects -Apiary was a very simple tool to use -had a lot of problems with AWS, tricky to get right -liked using Angular2 and TypeScript! Though I know a lot of groups used AngularJS because they didn't know the difference between the two -GitHub integration with Slackwas very useful -made it seem like the project was a real job Speakers Overall, all were good. Felt like some of the material was repetitive, but it was nice to get different points of views Quizzes some of them were pretty hard, but help understa

---

9. My only disappointment is that I was under the impression we would be learning more about software engineering in lecture than the semantics of Python.

---

10. Overall my experience in the course has been wonderful. I categorize you, Dr. Norman, and Mike Scott as essential instructors that all students should take while at UT. I won't go into details, but there are very few things that you can change to make it better. It is organized, thorough and I am excited to be in class from your enthusiasm alone. Thank you for the experience. I only wish I could have taken this class sooner and potentially TA'd afterwards.

---

11. Without internship experience, this course provided me with much needed insight into a developer career - especially an introduction into various tools and development practices.

---

12. I thought pylint was a little annoying since it has its own coding conventions that may not follow either your own or the company that you work for. Other than that I thought that the course was well organized. Thanks for a good semester

---

13. Not a fan of the of the class. Quizzes are too easy to mess up because we can't see the previous questions, we were given the impression that 5-6 quiz grades would be dropped, but it looks like only 1 will be. No way to evaluate your group membersfor IDB in a way that could affect their grade talk to Alison Norman for group evaluation ideas . Students are called out unfairly - everyone should be called on a proportional amount. I do not appreciate being made fun of when I answer a question,even if it's how you treat everyone. I think your policy about absences is RIDICULOUS. You praise a job as the highest honor, but don't allow for absences while we are interviewing. Not helpful that you think SWE is the only kind of job we should get

---

14. Overall I found this course very enjoyable, and probably one of the best courses that I have taken at UT. In particular I enjoyed attending the lectures. Personally I liked the way that Prof. Downing would call on students, and guide them to the correct answer instead of giving a rush of information. What I would have liked in this course would be to maybe scale back the python lectures a week or so, and cover a few of the tools that would be used in the final site. For example there was a lecture that covered database design, that would have been useful before I spent hours trying to manage a working, but terrible database for our final project.

---

15. Professor Downing is an excellent professor. I've had the pleasure of having him for OOP and SWE and in both cases the courses were excellent. He was always engaged in the classroom and reading blogs keeps him in touch with issues that students may have with the class or material. In a way this is an agile way of teaching the course! One aspect I did not enjoy very much was the weight of the exams on the overall class grade. Over all projects I spent over 50 hours working to demonstrate my ability with the material while tests were only 4 total hours. However, these were roughly the same amount of weight on the overall grade. It would be nice if tests weren't such a big piece of the pie. Thank you professor!

---

16. Collatz and Netflix are good projects to start on since they are not so advanced that they are intimidating to anyone who hasn't used Python before but can still provide good challenges to those at all levels of knowledge. There were a couple of instances where Piazza was used to inform the class about project requirement updates or important clarifications. This seemed a bit unfair because many students use Piazza when they are stuck but may not check it frequently if they are not having issues. A partner and I lost points on a project because we thought we were finished with the project and didn't see an important clarification that was posted after we submitted. In the future, any project updates should be reflected on the project page.

---

17. 1. I liked everything git, GitHub, pylint, unittest, coverage, pydoc, autopep8, Travis CI . The main thing I ended up not using was Docker. Instead, the school computers were acceptable and environment variables were done with python virtual environments. 2. Great introductory projects! I enjoyed them both. Sphere was a little weird but no complaints. Will try out Kattis. 3. I enjoyed the group project but sort of disagreed with proposals. It felt that without basing the database on media and the requirement of many attributes made it difficult to come with a unique and fun idea. It did force us to come with a good original idea, but this was arguably the hardest part. 4. Great exposure. Liked JPL in particular.

---

18. We should have atleast 1 excused absence when it comes to quizzes. I feel like quizzes could also be a little friendlier , almost everyone I've spoken to seems to have a poor average on the quizzes. It would be nice if IDB1 was introduced sooner and we were told to come up with ideas and groups earlier so we can focus on the creation of the website. It would also be nice if we could get a slightly better idea on what to expect on exams, there's alot of material and some if it might not makeit to the exam so being able to focus on specifics would be nice. Still overall I think this is a good and useful class.

---

19. Having taken OOP the beginning of SWE felt far too similar. I would have liked to have started with a different project. That being said the addition of Docker was an excellent idea. Comparing OOP and SWE again I miss the Google Doc format of notes you provided in last semester's OOP. I did not find the Netflix project to be particularly helpful. The main purpose seemed to be to increase our understanding of python dictionaries. My favorite speakers were Alex Menzies from JPL and Spiceworks. Alex gave a valuable talk about lessons learned and a neat demo. Spiceworks also had a lessons learned section. Other speakers were generic and talked mostly about their companies. I think that pin locked quizzes are unnecessary.

---

20. Please allow students to drop quizzes missed for interviews! Especially because this is a software engineering course! It's hard to have onsite interviews when you have quizzes MWF and can't miss any of those days

---

21. Learning pylint was great, but it would have been easier to learn the conventions for python syntax if the sample code provided also followed these conventions. especially code given in Collatz Also really enjoyed the talks from people at variouscompanies!

---

22. First project is a bit rough in terms of learning about setting up the production environment. I was left kind of confused as to what all the parts do specifically and what some of their industry alternatives would be, but it was a fun challenge.

---

23. My main conflict with this class has to do with the grade distribution. For the amount of time that we put into the projects, I'd prefer their net worth to be greater than the tests, which are just one of learning experiences that are already evaluated by the daily quizzes. Overall, I feel like the majority of my learning came from the projects, so I'd like to see where I learned the most also be where I'm evaluated the most.

---

24. Everything was great. Keep it up.

---

25. Using git Github, unittest, coverage, and Travis I feel will be extremely helpful in the future. Not sure how I feel about pylint. I would have like more detailed explanations of the non-group projects, especially with using the tools above, since they were our first projects so we were new to the tools. The group project was a really good experience. Flexible, funny, and intelligent group. Got to further dev my frontend skills, so it was very worthwhile. Wasn't too stressful at all. Speakers were good, if I had to rank them from most interesting to least JPL, Pariveda, Bloomberg, Spiceworks. Might want to encourage students to pay more attention be respectful, as there were many on their phones. Would like more quiz drops .

---

26. Dr. Downing was great! Wouldn't you agree?!

---

27. I like that I got some experience with continuous integration. I feel like the group project exposed me to many different tools needed to make dynamic website, but I do not feel like I mastered any of them. Most of the time was spent just trying to get the minimum requirements and features to work and move on to the next problem. Maybe there could be a piazza post next year where students could post tutorials or resources they found to be helpful. I liked that Pariveda and Spiceworks had given space to career advice in their presentations. I myself would prefer actual cash, so I can munch on more burgers from P Terry's.

---

28. Enjoyed the lectures. Enjoyed guest lectures. However, I quickly tuned out often since I already had an offer and were not interested in the companies. Wish the lectures and projects applied to each other more, especially for the website. Office hours were sometimes helpful depending on the TA. Wish the quizzes and test were more similar. Did not know what to expect for the first test. Appreciated knowing what to expect on the second tests. Website was not graded promptly.

---

29. I think Professor Downing is by far one of the best professors at UT. I only wish we had used Jenkins over Travis just to kind of "fit" in with industry standard. Aside from that, I had a blast!
