*** PROVISIONAL REPORT ***

```
UNIVERSITY OF TEXAS AT AUSTIN                  COURSE-INSTRUCTOR SURVEY          Fall 2015   DEPARTMENT COPY
Downing, Glenn P          C S373      50898    SOFTWARE ENGINEERING              Enrollment = 53
E100 EXPANDED                                                                    Surveys Returned = 49
```

| | NUMBER CHOOSING EACH RESPONSE | | | | | NO. REPLIES THIS ITEM | AVG. |
|---|---|---|---|---|---|---|---|
| | Str Disag | Disagree | Neutral | Agree | Str Agree | | |
| 1 COURSE OBJECTIVES DEFINED-EXPLAINED | 0 | 1 | 1 | 16 | 31 | 49 | 4.6 |
| 2 INSTRUCTOR PREPARED | 0 | 0 | 0 | 10 | 39 | 49 | 4.8 |
| 3 COMMUNICATED INFORMATION EFFECTIVELY | 0 | 0 | 0 | 13 | 36 | 49 | 4.7 |
| 4 STUDENTS ENCOURAGED-ACTIVE ROLE | 0 | 0 | 1 | 14 | 34 | 49 | 4.7 |
| 5 INSTRUCTOR AVAILABILITY | 0 | 0 | 3 | 14 | 32 | 49 | 4.6 |
| 6 COURSE WELL-ORGANIZED | 0 | 0 | 1 | 14 | 34 | 49 | 4.7 |
| 7 STUDENT FREEDOM OF EXPRESSION | 0 | 0 | 4 | 12 | 33 | 49 | 4.6 |
| 8 CLASS PARTICIPATION ENCOURAGED | 0 | 0 | 0 | 8 | 41 | 49 | 4.8 |
| 9 ENGAGING INSTRUCTION | 0 | 0 | 2 | 9 | 38 | 49 | 4.7 |
| 10 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT | 0 | 0 | 0 | 13 | 36 | 49 | 4.7 |
| 11 INSTRUCTOR EXPLANATIONS CLEAR | 0 | 0 | 1 | 11 | 37 | 49 | 4.7 |
| 12 GENUINELY INTERESTED IN TEACHING COURSE | 0 | 0 | 0 | 7 | 42 | 49 | 4.9 |
| 13 HELPFUL COURSE MATERIALS | 0 | 2 | 12 | 12 | 23 | 49 | 4.1 |
| 14 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS | 1 | 2 | 3 | 15 | 28 | 49 | 4.4 |
| 15 ASSIGNMENTS AND TESTS RETURNED PROMPTLY | 0 | 0 | 7 | 16 | 25 | 48 | 4.4 |
| 16 ASSIGNMENTS USUALLY WORTHWHILE | 0 | 1 | 1 | 16 | 31 | 49 | 4.6 |
| 17 STUDENT PERFORMANCE EVALUATED FAIRLY | 0 | 2 | 2 | 19 | 26 | 49 | 4.4 |
| 18 STUDENT PERCEPTION OF AMOUNT LEARNED | 0 | 0 | 0 | 20 | 29 | 49 | 4.6 |
| | Vry Unsat | Unsat | Satisfact | Very Good | Excellent | | |
| 19 OVERALL INSTRUCTOR RATING | 0 | 0 | 1 | 8 | 40 | 49 | 4.8 |
| 20 OVERALL COURSE RATING | 0 | 1 | 2 | 15 | 31 | 49 | 4.6 |
| | Excessive | High | Right | Light | Insuff | | |
| 21 STUDENT RATING OF COURSE WORKLOAD | 4 | 18 | 27 | 0 | 0 | 49 | |
| | 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 | 7 | 17 | 23 | 49 | |
| | __A__ | __B__ | __C__ | __D__ | __F | | |
| 23 PROBABLE COURSE GRADE | 14 | 25 | 10 | 0 | 0 | 49 | |

---

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: 18

---

1. You really need to be mentally prepared to take one of Downing's classes. This will probably be the first C that I've ever gotten in a course, ever.

---

2. Too stringent

---

3. Python stuff was the most engaging. XP and Refactoring was really dry to me. The Java details were cool too. I enjoyed the non-group project more than the group project, because for the group project some of the specifications for grading seemed somewhat arbitrary.

---

4. Greatest class, learned a lot, I see the course is still improving, please keep the good work!

---

5. The IDB projects were definitely the most informative part of the course. The open-ended nature of the project was great and allowed me to delve as deep as I wanted into the various tools we were using, as well as other tools that proved useful to the implementation of our site  nginx, uwsgi, marshmallow, among other things .  That said, my only complaint with that project was the arbitrary nature of the unit tests. The models, realistically, were the part of the project that required the leastamount of testing. My group did some additional unit testing for API calls which was much more helpful in catching errors early. Unfortunately I don't have any suggestions for improving the unit test requirements, but that was my impression of them.

---

6. I really liked how the class provides industry-focused computer science education, since most of the classes in the department are theory-focused. In addition, the tools and projects are very useful resource for the interviews.

---

7. I wish the course better covered refactoring, development styles like waterfall, scrum, etc. and revision control. I think it would be better to have projects that made us use refactoring, or maybe make our own revision control, instead of making a website. I just didnt get much out of the website except that sqlalchemy is the worst thing to ever exist. Also since we are using sqlalchemy, why spend time teaching sql? We did not have to use that in the project at all. That time could be better spent learning more about the java bindings and refactoring things we covered at the end, which I thought were very interesting but barely touched on. At the end of the course I feel that I do not know what software engineering is, but I know python.

---

8. Great projects.

---

9. Professor requried too many contrived parts of projects  e.g. using somebody else's api in an arbitrary way or using some unspecified amount of embedded media  and didn't have any particularly challenging or interesting requirements on the main web development project. It would have been more interesting to have us demonstrate use of design patterns or work on some more focused project like building an ETL tool or data visualization or implementing a security system in our site, etc.  Not enough preparation material was provided in order to do well on the tests. Practice tests would have been really nice and educational.

---

10. I loved this class. The first assignment really helped me get more used to new tools, and being able to choose a topic for the larger project has awakened a lot of passion in me. I love Pokemon and so do my other team members, so we were into the project a lot more than we would have been otherwise. I want to continue the website and make its future evolution a personal project. Thank you so much for giving me this opportunity, Prof. Downing! The speakers were interesting and I really enjoyed learning about how to get a website going. I liked the class participation and the quizzes were at just the right difficulty. I think you should document how you teach because I honestly feel it will help the education community by a lot! Best class.

---

11. I have taken both SWE and OOP and have found these courses to be two of the most rewarding classes in all of CS.

---

12. Software Engineering was a great class. I learned a lot of stuff that I was previously unaware about, and it also sparked my interest in Full Stack Development. Downing is a great professor and one of the best teachers I have had so far in my academic career.

---

13. Professor Downing's classes are my favorite classes because he really tries to teach you skills that apply directly to the workplace after graduation. They are incredibly tough, but they really make you push yourself. I like that he tries to make sure his course is up to date with current technologies. His supplemental resources and his website github are incredibly useful as well. I do find him very intimidating however, I think because of how tough he is on us. It makes it difficult  for me anyways to talk to him outside of class. I'm also always dreading being called on in class and it makes me feel like I'm the only dumb one in the room if I don't know the answer.

---

14. I wish that this class had focused more on the design patterns and refactoring that we did towards the end of the class. I think they're an invaluable tool that aren't very easily learned on one's own. This class felt like OOP in python, except withan emphasis more on one of python's strong points - making a website quickly. Downing is obviously a phenomenal professor, and I learned a lot about python that I didn't already know, but I think it would have been more valuable for me if we spent more time on theory and less on language-specific information.

---

15. Thank you for a great semester!

---

16. I really enjoy having Prof. Downing teach about coding. He keeps things simple enough to understand while at the same time keep a high standard for his students which I think is valuable for a teacher. So while the quizzes and tests can sometimes behard but fair, he keeps us on our toes by forcing us to not to remember questions and answers like the conventional classes but to figure out a solution based on the little building blocks he teaches us.

---

17. Great teaching style and projects.

---

18. Really great class, I enjoy your style of lecturing. I liked all of the tools we used as well. If I had to be critical of something, it would be that the tests should be weighted less, and the IDB projects should be weighted more. I don't write codevery well with pen and paper  or typing in any environment where I can't refer to something else . I feel like IDB projects could be used to test knowledge of the more complicated material  like inheritance, abstract classes, interfaces, refactoring, etc. by having to submit a "snippet of code" that uses it or something. Thanks!

---