Individual Instructor Report Spring 2023 Version A for C S 373 - SOFTWARE ENGINEERING (52410) (Glenn Downing)

Project Title: **Instructor Course Evaluations Spring 2023**

Courses Audience: **60**
Responses Received: **48**
Response Ratio: **80.0%**

Report Comments

Results were collected during the implementation of a new course evaluation system.

Guide to the Interpretation of Course Evaluations at UT Austin
The goal of course evaluation process at UT Austin is to drive teaching excellence and to support continuous improvement in teaching and learning experiences. The two sets of scales used for core evaluation questions and the associated weights are:

Strongly Agree (5)
Agree (4)
Neutral (3)
Disagree (2)
Strongly Disagree (1)

Excellent (5)
Very Good (4)
Satisfactory (3)
Unsatisfactory (2)
Very Unsatisfactory (1)

The Mean is calculated by adding all of the weights for a single question and dividing by the number of respondents. The course workload question is not averaged.

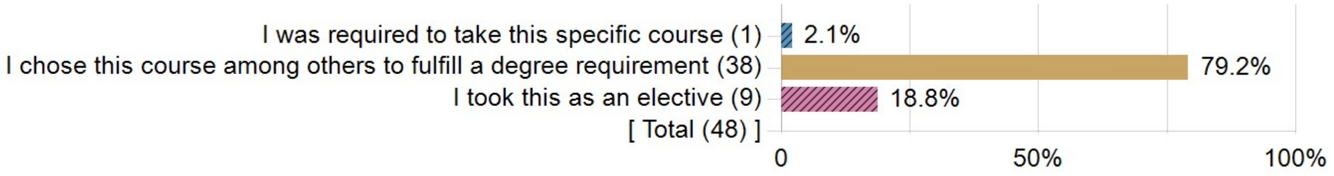
The number of students (e.g. Respondents) marking each option is reported for each of the items. These frequency distributions provide information about the level of student ratings and the spread and shape of the class distribution of responses. The distributions thus provide a picture of student perception of a course.

Course evaluations provide snapshots of student perspectives on their course-level learning experiences. Most experts on teaching evaluation advise that no one method gives the complete picture of an instructor's teaching effectiveness; multiple and diverse measures, on multiple occasions, are advised to give a full picture of the teaching effectiveness of a particular instructor. Moreover, other factors, such as size of class, level of the class, and content of the course, can cause small variations in the ratings. Therefore, student perspectives for a particular instructor or course should be interpreted as a snapshot, and not as providing complete information on the teaching effectiveness of that instructor.

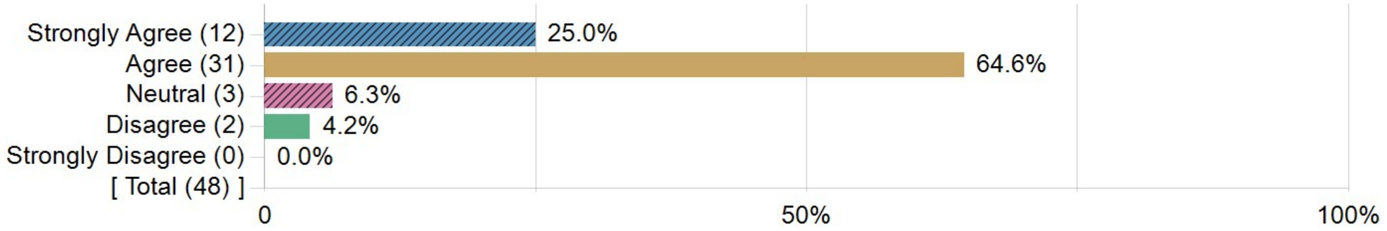
Creation Date: **Thursday, May 11, 2023**

Course Questions

Why did you take this course?

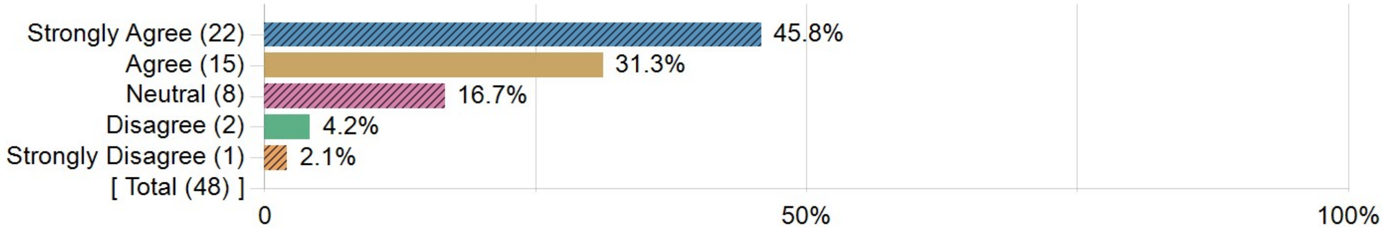


During this course, I gained a deeper understanding of the subject matter.



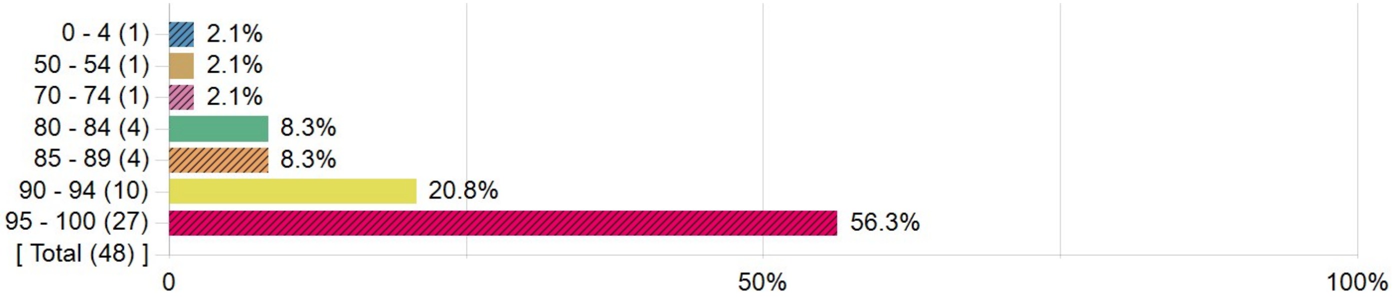
Statistics	Value
Mean	4.10

The course was well organized.



Statistics	Value
Mean	4.15

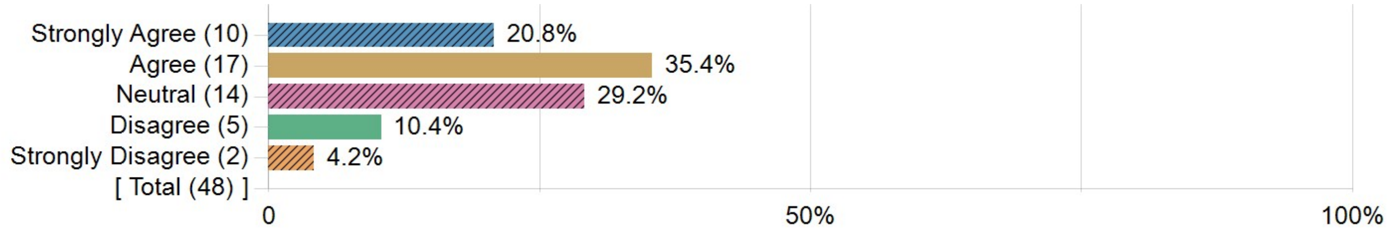
Overall, approximately what percentage of the course meetings did you attend or complete (online, in person, or asynchronously)?



Statistics	Value
Mean	90.00

The course assignments improved my ability to express my ideas in writing. (Flag Question)

The course assignments improved my ability to express my ideas in writing.



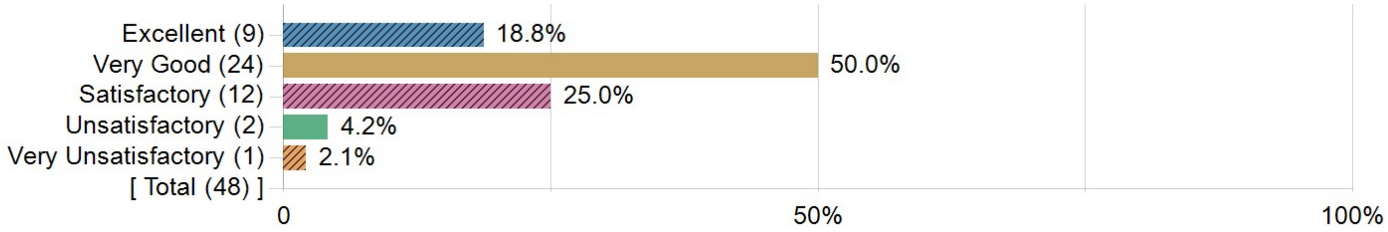
Statistics	Value
Mean	3.58

Instructor Questions

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Responded	Mean
The instructor clearly explained the course objectives and expectations.	54.2%	43.8%	0.0%	2.1%	0.0%	48	4.50
The instructor fostered an inclusive learning environment.	50.0%	41.7%	6.3%	2.1%	0.0%	48	4.40
The instructor effectively explained the concepts and subject matter in this course.	35.4%	52.1%	4.2%	4.2%	4.2%	48	4.10
The instructional techniques kept me engaged in learning.	43.8%	41.7%	10.4%	4.2%	0.0%	48	4.25
The instructor checked for student understanding of the concepts presented in the course.	56.3%	35.4%	8.3%	0.0%	0.0%	48	4.48

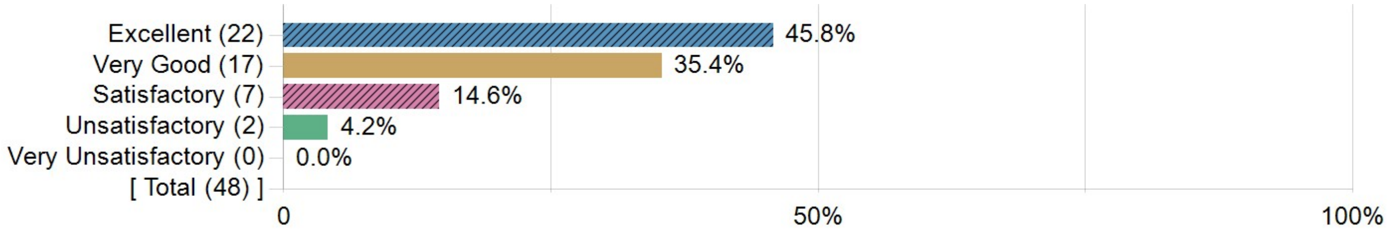
Overall Questions

Overall, this course was



Statistics	Value
Mean	3.79

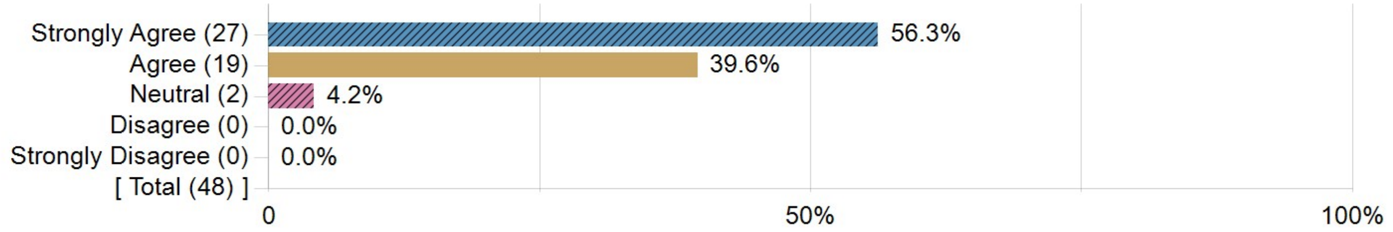
Overall, this instructor was



Statistics	Value
Mean	4.23

College, School, or Unit Questions

The instructor fostered a classroom environment in which all students could feel free to participate fully.



Statistics	Value
Mean	4.52

Comment Questions

Identify aspects of the course that were the most effective in helping your learning.

Comments
Aspects of the course that were the most effective in helping my learning consisted of the daily quizzes, exercises, and the project. The daily quizzes made sure I didn't fall behind in the class. My favorite part about the class was by far the IDB project, because I felt like I learned the most out of it, and I was able to create something that was impactful and I felt proud of/accomplished.
How classes are organized. It is creative and effective.
The projects were the most effective in helping my learning as well as having access to past students' repositories so that my group and I can get better ideas for implementing certain components.
website was helpful with organization
I liked the lecture style and how Professor Downing treated the lecture like conversations with individual students. However, sometimes it could feel like the conversation dragged on and it would be easy to lose focus.
The most effective aspect of this class was the material that we learned in class and were quizzed over. I felt like I gained a deeper understanding of Python, SQL, and new frameworks that we used in class.
The "cold call" system where the professor calls on students randomly to answer questions.
Hands on project, team based assignments.
Working with teammates really helped me learn new things but the type of teammates you can get are hit or miss. Cold calling was a great way for me to ask questions and stay engaged in class.
The class projects
I hate to say it but the in–class quizzes and exercises really make it imperative to actually pay attention and focus on the class material. The cold–calling especially, you don't want to get caught off–guard and sound like an idiot. But the structure of the class is great and everything is set up so you know what to focus on at every step. The papers we read were interesting and informative, I liked that they were authored by real people in the industry and touched on real practical situations and techniques.
Lots of examples of concepts in class
I will say that the entirety of this class was useless, we only learned a lot from this class cause we all had to learn everything on our own outside of class.
The practicality of the lesson material and project goals and structure
Exercises were amazing
The semester group project was pretty realistic and introduced me to a bunch of web dev tools that should be useful in my career. However, it was paced poorly, with the majority of the work being on one phase (phase II).
The teaching method during lecture made sure that I was focused and engaged.
Projects and lectures
I think the format of the lectures was great and helped me to retain the information.
Daily quizzes. And cold–calling
Projects outside of class
The in–class quizzes forced me to frequently study the lecture content, which resulted in me truly learning what was covered in the lecture.
Perusal notes
The engaging lectures, and open–ended projects.
Cold Calling helped you stay engaged.
I liked the in–class exercises. They were difficult but not too hard and had an appropriate time limit.
Cold calling and the quizzes helped me the most in learning.
I liked the projects for sure. I am a student who learns by doing, and being thrown into the deep end with these phases has made me learn a lot about how web applications really work.
I think the pressure of having to build a full–stack application taught me so much I wouldn't have learned otherwise about full–stack development.
Cold calling – made sure that you knew what the course content was about. Also enjoyed the projects and the fact that it was a lot of self teaching, made me learn more.
The lecture format was helpful for learning the content
I found the lecture notes to be really helpful as they required you to pay attention in class and helped a lot with understanding not

Comments
having to focus on your own notes
I liked the daily in class quizzes as opposed to exams or midterms. It forced me to look at the notes before every class, which is something I would not normally do if it was an exam–based class.
quizzes
The lectures were very effective
Honestly probably the exercises, they were a fun and collaborative application of the lecture material. The cold calling was also fun.

Identify the aspect of the course that you found most challenging, why you found it was challenging, and suggest one thing that could be done to help future students meet that challenge more effectively.

Comments
I found the communication between the TAs and the professor to be limited. On multiple occasions regarding the IDB project, I would reach out to the TAs and what they told me did not align with what the professor would tell me. I would find it to be very confusing, because I didn't know who's instruction to follow. Additionally, the IDB rubric on the course webpage was unclear and not very detailed. I appreciated the google doc project rubric a lot more, because it was more clear and detailed. I think that going forward, the course webpage rubric should be based on the existing google doc or the google doc rubric should be used solely.
Quizzes.
The projects were the most challenging because there was no guidance at all in understanding new tools. One thing that can be done is to have at least one lecture that goes over the organization of the IDB project and explains at a high level the purpose of each of the software development tools we are supposed to do.
the quizzes were really bad for me
I thought the fact that the lectures were so disconnected from the projects was very challenging. It made it difficult to learn the material necessary for the projects.
The most challenging part of this class was definitely the IDB project. There was nothing that we learned in class that prepared us for the project. There were so many things to learn on our own that would have been so much more helpful if SOME initial tips or guidance were given in class.
The lecture material had nothing to do with the project, which was a major part of the course's grade. I would make the lectures more relevant to the project, such as lectures on React and Javascript.
Lecture content was disconnected – would be nice to cover some more technical aspects of project in class.
If you get bad teammates it feels like you are set to fail in class. There seems to be not much repercussion for not contributing to work and its very frustrating that that is the case. Also it is extremely hard to keep track of what your grade is in the class. Because it is not based off of percentage points you can suddenly drop from an A to a C in a matter of days because of 1 or 2 small things you forgot.
The quizzes. I think a couple of minutes added to the timed quizzes might help answer the questions after thinking about the question thoroughly
More balanced grading system. People don't typically miss even one of their projects but get penalized much too heavily for the other requirements.
The IDB project not being related anything to the lectures kind of messes with me still, I wish there was more to cover that project in class. My team had no real web–dev experience and having to learn all of it as we did the project proved to be challenging to me at least.
Learning some parts on my own on web development
The quizzes were the hardest thing for me, four minutes to figure everything out is tough.
In terms of grades, the attendance quizzes, so make sure to study so your only misses are inevitable absences, and in terms of effort, learning and completing the project (especially phase 2) on your own, so make sure to do it incrementally with good timing.
More time on quizzes and make them open note
The quizzes at the beginning of every lecture, I felt, were tricky and extremely hostile to anyone who hadn't brute–force memorized the previous lecture. Instead of directly addressing broad, main ideas, each question incorporated multiple specific aspects of what we learned previously in ways that were hidden or often seemed intentionally misleading. On top of that, the quizzes had very strict time limits, which again reinforced brute–force memorization rather than on–the–spot problem solving or logical thinking (and even then sometimes the time felt like it wasn't enough – three minutes? really?). I felt like I had been setup to lose when I realized at the end of the semester that, despite doing extremely well in every other part of the class, these trick quizzes would mean I get a C overall. I think either these quizzes should be reworked to be more forgiving, or the grading scheme should be reworked to de–emphasize them.
Projects are tough with random teammates

Comments
I think the project is really challenging to begin with and suggest that resources for getting started be released earlier to help future students.
I could use some extra motivation in learning new technologies in the projects.
Class content had little to do with subject matter
I found the projects pretty challenging, mainly because I was thrown into a project I had no requisite background knowledge for. I feel that teaching the basics of JavaScript or React, and/or relating the lecture content to the project content a bit more would have helped a lot.
The fact that the semester long project and the content we went over in class had 0 overlap
The exercises. The fact that the exercises are pop quizzes makes them much harder, because you don't know when to take extra care and study more the previous day, which makes them especially hard when factoring in specification grading
The quizzes were very challenging at times since the notes sometimes did not reveal intricate meanings.
I found the daily quizzes the most challenging because of the short time limit. 1–2 extra minutes on the quizzes could go a long way.
The quizzes were the most challenging, but it helped when I would study before class.
I found the projects to be challenging because every tool was new you feel me. I was switching between languages, switching between frontend and backend development, and it was just hard getting used to everything you feel me.
Ironically, learning full–stack development was a double edged sword. The reason being was that most class material was not helpful for the development of our full–stack project. It was valuable in other ways, but we had to learn a lot of tools on our own to finish the projects.
Because many of us were new to software engineering, some of the phases were difficult and the TAs were not super responsive.
Figuring out how to get started with IDB and set everything up with no guidance
I found the disconnect from python and the projects we were assigned to be a little challenging. Maybe introduce materials and homework outside of class to help with the projects instead of a paper and blog.
I found it challenging that there was no connection between projects and the lectures. I also found the grading scheme to be really discouraging toward the end of the semester. Because in–class quizzes were difficult, my quiz score brought down my entire grade.
I found the projects to be challenging since I did not have any experience with backend development. I understand that the point of this class is to simulate a real–world software engineering job, but it would be nice to have a couple more resources and a couple lectures on helping us get started.
quizzes
The project was very challenging because we didn't know how to use most of the technologies and we weren't given much direction on how to use them either.
Downing himself is GREAT but the class right now is not. I go into a lot of this in my final blog post for the class. But essentially the disconnect between the lectures and the projects, I find really problematic and it really hindered my learning for the course. I think this disconnect is deliberate, because he wants us to have to self–teach new tools. But I think the way it's done, is not representative of the workplace or industry. Because even in industry there will probably be some sense of familiarity, yet with the projects, there was none at all. So for stragglers in CS like me, I just struggled too much to the point it felt unfruitful and just not "worth it." Now I think I just have really shaky foundations or knowledges of all the tools, because I had to self–teach. And self–teaching is easier or harder depending on the student, their commitments to other things (organizations, job etc) and their experience level. So I think this class favors (or benefits more) those who have already had internships, because they come into the class inherently more prepared. But I don't have experience like that, so this class just left me behind.