

STUDENTS' RATINGS/COMMENTS ON MODULE

Faculty:	SCHOOL OF COMPUTING	Academic Year:	2013/2014
Department:	COMPUTER SCIENCE	Semester:	1
Module:	SOFTWARE ENGINEERING - CS2103T		

Items Evaluated	Module Avg Score	Nos Responded
Overall Opinion of the module.	4.424	66
Expected Grade for the module.	4.409	66
Difficulty Level of the module.	3.818	66

QNSCORE	5	4	3	2	1
Qn 1: Overall Opinion of the module.	Excellent	Good	Satisfactory	Unsatisfactory	Poor
Qn 2: Expected Grade for the module.	A	B	C	D	F
Qn 3: Difficulty Level of the module.	Very Difficult	Difficult	Average	Easy	Very Easy

Frequency Distribution (Qn 1: Overall Opinion on the module.)

ITEM\SCORE	Nos. of Respondents(% of Respondents)				
	Excellent	Good	Satisfactory	Unsatisfactory	Poor
Module	34 (51.52%)	26 (39.39%)	6 (9.09%)	0 (.00%)	0 (.00%)
Module at Same Level (Dept)	178 (22.62%)	365 (46.38%)	200 (25.41%)	35 (4.45%)	9 (1.14%)
Module at Same Level (Fac)	210 (21.38%)	455 (46.33%)	254 (25.87%)	46 (4.68%)	17 (1.73%)

Frequency Distribution (Qn 2: Expected Grade for the module.)

ITEM\SCORE	Nos. of Respondents(% of Respondents)				
	A	B	C	D	F
Module	32 (48.48%)	31 (46.97%)	2 (3.03%)	0 (.00%)	1 (1.52%)
Module at Same Level (Dept)	277 (35.60%)	397 (51.03%)	87 (11.18%)	13 (1.67%)	4 (.51%)
Module at Same Level (Fac)	333 (34.26%)	521 (53.60%)	96 (9.88%)	16 (1.65%)	6 (.62%)

Frequency Distribution (Qn 3: Difficulty Level of the module.)

ITEM\SCORE	Nos. of Respondents(% of Respondents)				
	Very Difficult	Difficult	Average	Easy	Very Easy
Module	5 (7.58%)	44 (66.67%)	17 (25.76%)	0 (.00%)	0 (.00%)
Module at Same Level (Dept)	131 (16.60%)	412 (52.22%)	221 (28.01%)	20 (2.53%)	5 (.63%)
Module at Same Level (Fac)	149 (15.16%)	489 (49.75%)	312 (31.74%)	26 (2.64%)	7 (.71%)

The best aspects of this module are:

1. -
2. Incremental workload and very hands-on.
3. Hand on experience of building a software for the first time.

4. Learning good practices of software engineering that could be applied to all future and current modules taken. Really helpful!
5. -Ample resources given, module plan is very structured
6. I tend to apply what i have learnt immediately in the product, and helps me to learn these concepts.
7. How inherently applicable it is to us as computing students. At the end of it all, I feel enlightened and confident I will be a better coder.
8. Learning a good variety of software engineering stuff.
9. Building of something useful and the team feels proud of.
10. The balance between theoretical and practical knowledge.
11. Very enjoyable. Provide a lot practical knowledge. Attend the lecture feel very differently. It is like a de-stress period on a Friday and good way to start TGIF.
12. I have so much fun in this module. I've learned so much in this module as compared to others, and likely to be useful from this point onwards.
13. The tutorials at the active learning lab is refreshing. Every week, students are tasked to present their homework. In this way, students are able to learn independently. The project in CS2103T builds up every single week, the students are required to submit an iteration that is better than the previous. Students decide what goes into the next iteration instead of Dr Damith enforcing certain restrictions. This creates the opportunity for students to relate to real life scenarios.
14. The lessons it teaches us that better prepare us for future coding projects
15. Getting to work on a project with a team.
16. Software engineering is fun.
17. The correct way of handling a new project. From planning to coding to launching of the product.
18. Very applicable for programmers.
19. Good code practice
20. Interesting lectures
21. what i learnt in lectures and the slides. Slides are so good, together with the lecture notes. Very fun to learn.
22. teaching how to have better code structure, better team attitude, big companies' product concepts
23. Nil
24. The project which gives some freedom in what we want to do
25. Interesting approach to teaching software engineering. Engaging and exposed me to the many different concepts and terminology in the area of software engineering. Encouraged me to learn more on my own.
26. Engaging lectures, and useful content.
27. Quite interesting
28. Effective lectures
29. Contents are highly relevant.
30. I learn the basic process of how to design and implement a product during the project.
31. Learn a lot of soft skills from the lecturer and project practices
32. The tutorials and lectures are closely related. The timeline is very clear at the beginning. Although the workload is heavy, it helps a lot.
33. Good lecturer and tutors. Very useful for students.
34. -
35. the team project
36. It provides hands-on opportunities for students to try out what they have learned.
37. The project gives me a wonderful experience! Love to teamwork with others and complete a product which can benefit some users
38. The course teaches me many practically useful techniques in software engineering. The project gives me a great opportunity to experience how a team create something.
39. Learning to apply all the theory of software engineering to the concurrent project that we do for the module.

This module could benefit most by:

1. -
2. -
3. -
4. -Module delivery is extremely fragmented, resources are all over the place, and there are many types of resources. For example, announcements on both module page and IVLE are confusing.

Lecturer response: Everything related to the module can be found in one of these two places: the Handbook page (organized by topic) or the Schedule page (organized chronologically). The announcements shown in the website is the same IVLE announcements page. I'm surprised you didn't realize that :-p.

5. Having a slightly smaller scale final product / less workload , i think this module is honestly quite heavy in terms of how much hours is required to be put in. Our group spent quite a fair bit of time on the final product.

Lecturer response: Yes, the project is heavy, but lighter than previous semesters. This is a 60% CA module. It is inevitable that the project is heavy. In comparison to level 3-4 project modules, our project is much lighter :-)

6. Some of the deadlines are very close together. One thing I also noticed is how the final product places pretty significant emphasis on creating a good user experience. However, the lectures only cover this aspect briefly. Perhaps in the future, slightly more emphasis can be put on this aspect? I see great benefits in adding this to the curriculum - It will break the monotony of the drier parts of the module, and is something that is very relevant and something we can relate to, both as coders and as students.

Lecturer response: Yes, the user experience is important. This module requires you to apply common sense and intuition in designing the user experience. Deeper theories of interaction design are left to higher-level modules.

7. Starting the project earlier.

8. -

9. I heard build automation was covered in the past. Perhaps it could be brought back. The final lecture seemed more suited for week 13 than 12.

Lecturer response: We do touch on build automation in handouts. No it wasn't covered even before. Week 13 is for project demos. That's why we have to finish lectures earlier.

10. -

11. -

12. reducing workload

13. Lightening the workload

14. -

15. Check for mistakes in handouts and correct them before uploading.

Lecturer response: Already done. If you notice further mistakes, please let me know.

16. The project is crazy though, takes my time and burns it awaaaayyy

17. -

18. -

19. The pace of the class is quite fast in the middle few weeks where there is a lot of content to cover on OOP etc. Maybe split it up further so that there is a better balance in workload and learning content over the weeks. The first few weeks tend to be a bit slack.

20. making lectures more efficient.

Lecturer response: We use the 'flipped classroom' approach where there is less content delivery in class. Instead, we use the classroom to discuss about WHY we learn what we learn and HOW they relate to practice. This appears inefficient, but it is more effective in transforming your minds, which is what a university education should do.

21. -