

JHU - Krieger School of Arts & Sciences / Whiting School of Engineering
ASEN.2019.Fall

Course: EN.601.417.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 16/16 (100.00 %)

1 - The overall quality of this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.81
Weak	(2)	0	0.00%		
Satisfactory	(3)	0	0.00%		
Good	(4)	3	18.75%	■	
Excellent	(5)	13	81.25%	■	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
16/16 (100.00%)			4.81	0.40	5.00

2 - The instructor's teaching effectiveness is:					
Yair Amir					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.75
Weak	(2)	0	0.00%		
Satisfactory	(3)	1	6.25%	■	
Good	(4)	2	12.50%	■	
Excellent	(5)	13	81.25%	■	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
16/16 (100.00%)			4.75	0.58	5.00

3 - The intellectual challenge of this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.81
Weak	(2)	1	6.25%	■	
Satisfactory	(3)	0	0.00%		
Good	(4)	0	0.00%		
Excellent	(5)	15	93.75%	■	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
16/16 (100.00%)			4.81	0.75	5.00

4 - The teaching assistant for this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.88
Weak	(2)	0	0.00%		
Satisfactory	(3)	0	0.00%		
Good	(4)	2	12.50%	■	
Excellent	(5)	14	87.50%	■	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
16/16 (100.00%)			4.88	0.34	5.00

Course: EN.601.417.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 16/16 (100.00 %)

5 - Please enter the name of the TA you evaluated in question 4:	
Response Rate	15/16 (93.75%)
<ul style="list-style-type: none"> • Sahiti • Sahiti Bommareddy • Sahiti Bommareddy • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti Bommareddy • Sahiti 	

6 - Feedback on my work for this course is useful:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Disagree strongly	(1)	0	0.00%		
Disagree somewhat	(2)	0	0.00%		
Neither agree nor disagree	(3)	1	6.25%		
Agree somewhat	(4)	0	0.00%		
Agree strongly	(5)	15	93.75%		
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
16/16 (100.00%)			4.88	0.50	5.00

7 - Compared to other Hopkins courses at this level, the workload for this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Much lighter	(1)	0	0.00%		
Somewhat lighter	(2)	0	0.00%		
Typical	(3)	1	6.25%		
Somewhat heavier	(4)	4	25.00%		
Much heavier	(5)	11	68.75%		
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
16/16 (100.00%)			4.63	0.62	5.00

Course: EN.601.417.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 16/16 (100.00 %)

8 - What are the best aspects of this course?

Response Rate	15/16 (93.75%)
----------------------	----------------

- professor's knowledge and experience and enthusiasm, interesting material, practical coursework
- Enthusiastic and knowledgeable professor and extremely helpful TA. I learned a lot from them
- We had a chance to explore the cutting edge research areas in distributed systems.
- Class projects have a good challenge level. Professor is very excited about the material and makes it fun to learn. I also liked the theoretical assignments.
- Yair is probably the best teacher I've had at Hopkins, you can tell he cares deeply about his students, with the time he puts into going through the design of each project with each group individually and giving feedback on them. Not only that, but his lectures are engaging (I especially liked the ones toward the end of the semester that were very interactive), and his expertise in the field really comes across and is helpful. I think any CS major hoping to do systems should take this course, even if they don't end up in the specific field of Distributed systems.
- The contents are very intellectually challenging. The professor and TA are very helpful during office hour.
- The exercises are extremely challenging but intellectually stimulating. You learn a lot of things, both theoretically and practically, about distributed systems and how to build them in real life. The design meetings are super helpful. It saved us a lot of time from implementing things that are suboptimal. This is something I think many CS courses should have. The student should have both the right theoretical understanding and the practical skills to translate the understanding into code. Having a design meeting makes both the theoretical and practical training more effective. Both Yair and Sahiti are very kind and always made themselves available for the students. They are always very responsive on piazza and are very helpful. The feedback on the assignments are designed to help us improved. The final project meeting is especially instructive.
- The best aspect of the course was the knowledgeable and enthusiastic professor. He also never hesitated to help a student. It was also great to hear about his real world experience in the field.
- The programming projects helped me take abstract ideas discussed in class and put them into practice.
- Dr. Amir was a fantastic teacher. He truly cared about the students and making sure we were learning the course material. He and Sahiti put in so much time to make the course a success, and he curated the content so that it was related to what we were all interested in. I learned so much!
- Very interesting topic and thought provoking presentation of relatively unknown issues
- The professor is extremely engaged in this course topic. He is a truly a leader in this field and no one could teach this class as well as he can. The class teaches a new way of thinking about this material and is truly incredible experience. I can't imagine going through my four years at hopkins without taking this class.
- Professor Yair is very responsible. The way they are teaching the course is rare but rewarding. TA is great!
- There's a strong focus on design and the staff is available to help.
- Working on the projects gave a lot of insight into working on real distributed systems.

9 - What are the worst aspects of this course?

Response Rate	15/16 (93.75%)
----------------------	----------------

- time needed per project (easily 20+ hours)
- N/A
- N/A
- Hard to complete the final project in time without having to work over Thanksgiving break.
- There are none
- Homework requires a lot of time and effort. It is also hard to figure out the exact goal of the homework from the project description. Need to read the description for several times and consult the professor and TA a lot.
- The lectures could be more organized and tied together in a more clear way. Compared to the design meetings and the exercises, I feel I gained less from the lectures.
- The worst aspect of the course was the pacing. It may have been beneficial to spend more time on the later theoretical topics rather than the earlier ones, as the later theory was a bit harder to grasp.
- Sometimes, the lack of feedback on progress in the course made it difficult to know how I was doing. Perhaps feedback could be given more frequently, e.g., in the form of more frequent homework assignments. However, the feedback on the programming projects was very thorough.
- I can't think of anything!
- very high workload
- The course has really poor pacing. There are only 11 "lectures" but each of those topics can take 1-5 days of class. This made it hard to keep track of where we were in the material on a specific day. Also the course had major timing issues. All of the projects were given only 1-3 weeks with at least half of the time dedicated to design. This was never truly enough to complete all of the parts of the project that were asked for especially as an undergrad in other courses. There were large periods of time with no work and then everything would get piled on at the same time. There was not an assignment until about 3 weeks into the course and then we got two at the same time.
- No curve...
- There isn't a very cohesive progression throughout the semester. It felt like many different solutions were presented to common problems, but it didn't build on previous material.
- Very high workload that can sometimes be hard to balance with many other courses.

Course: EN.601.417.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 16/16 (100.00 %)

10 - What would most improve this class?

Response Rate	13/16 (81.25%)
---------------	----------------

- n/a
- Maybe more detailed slides and supporting materials would be beneficial as I don't find a lot of resources online to review some of the materials covered in class
- It's almost perfect.
- I would specify the final project a bit more.
- Making expectations for how the project should be designed and presented more clearly. I was under the assumption that the final project would involve more of a presentation rather than just a read through of the design document.
- Make the project description more clear. Also make more Piazza posts public. It'll help students know if they understand the course/project like most of other classmates do, especially in the class with students from different levels.
- Making it 6 credits :) (in all seriousness, the workload of this class is not of a 3 credit course)
- I think it would be great if there could be more lectures about the real world applications. For example, the lecture with John was great.
- For some lectures, I think that a blackboard-style (rather than PowerPoint) format would be better. Sometimes simpler is better, and I have found (in general) that instructors who use the blackboard are more successful at taking complex ideas and reducing them down to simple examples. I think this style of lecture would be effective in combination with the PowerPoint. E.g., start out by explaining a scenario that motivates the need for the algorithm on the whiteboard, describe the algorithm on the blackboard, talk about potential downsides on the whiteboard, then switch to PowerPoint to show graphs showing the performance of the algorithm in practice.
- More even spacing of the homework with less overlap and more time to work on each assignment.
- Don't know.
- More theoretical assignments would have been better. Also stricter timelines. The extensions weren't very useful with other workload.
- A "late days" system so that if an assignment in this class conflicts with an assignment or test in another class there is an option to submit late for some assignments.

11 - What should prospective students know about this course before enrolling? (You may comment on any aspect of this course such as assumed background, readings, grading systems, and so on.)

Response Rate	15/16 (93.75%)
---------------	----------------

- prior coursework in systems or networks recommended
- Huge workloads. Needs a lot of time to debug
- Don't enroll this class with other classes with heavy workload.
- Useful to have taken Computer Networks and also have a decent understanding of C.
- It is intense, but worth it. However if you are not comfortable working in C, it might be too much work.
- Be prepared to spend a lot of time in projects and theoretical assignments. Start early and allocate more time in design before implementation.
- Be ready to spend lots and lots of hours on the assignments. But when you get stuck, don't be afraid to reach out for help. It is often much more effective than spending lots of time implementing something that is suboptimal. This class is probably one of the most difficult the department has to offer. You will benefit from having an upper-level system course before: it will help you get used to the way of thinking as well as catching strange bugs. Being solid in C is also tremendously helpful.
- Prospective students should know that the course is extremely challenging but is quite rewarding.
- Strong programming skills in either C or C++ are necessary.
- Definitely have a strong background in c/c++ !
- be very familiar with C/C++
- This course is one of the most incredible courses I have ever taken. The professors is at the forefront of this field and has been working this material for all of his career. He puts a lot of effort into running the course and expects an equivalent amount of effort out of the students who take his course.
- You will be code heavily in C, which is not fun :)
- This is a research course. I would not recommend it for general knowledge (take OS instead) but if you want to work on building distributed systems it will be helpful. I don't think this course will be very useful if you want to learn how to use microservices, which will be a big area that's a hole in the CS curriculum.
- A strong C/C++ background is necessary and the willingness to put in a lot of work. The projects however are very rewarding.

JHU - Krieger School of Arts & Sciences / Whiting School of Engineering
ASEN.2019.Fall

Course: EN.601.617.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 13/15 (86.67%)

1 - The overall quality of this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.54
Weak	(2)	0	0.00%		
Satisfactory	(3)	1	7.69%	█	
Good	(4)	4	30.77%	███	
Excellent	(5)	8	61.54%	██████	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
13/15 (86.67%)			4.54	0.66	5.00

2 - The instructor's teaching effectiveness is:					
Yair Amir					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.54
Weak	(2)	1	7.69%	█	
Satisfactory	(3)	0	0.00%		
Good	(4)	3	23.08%	███	
Excellent	(5)	9	69.23%	██████	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
13/15 (86.67%)			4.54	0.88	5.00

3 - The intellectual challenge of this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		4.77
Weak	(2)	0	0.00%		
Satisfactory	(3)	0	0.00%		
Good	(4)	3	23.08%	███	
Excellent	(5)	10	76.92%	██████	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
13/15 (86.67%)			4.77	0.44	5.00

4 - The teaching assistant for this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Poor	(1)	0	0.00%		5.00
Weak	(2)	0	0.00%		
Satisfactory	(3)	0	0.00%		
Good	(4)	0	0.00%		
Excellent	(5)	13	100.00%	██████	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate			Mean	STD	Median
13/15 (86.67%)			5.00	0.00	5.00

Course: EN.601.617.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 13/15 (86.67%)

5 - Please enter the name of the TA you evaluated in question 4:	
Response Rate	12/15 (80%)
<ul style="list-style-type: none"> • Sahiti Bommareddy • Sahiti • Sahiti • Sahiti Bommareddy • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti • Sahiti Bommareddy • Sahiti Bommareddy 	

6 - Feedback on my work for this course is useful:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Disagree strongly	(1)	0	0.00%		<p>4.54</p>
Disagree somewhat	(2)	1	7.69%	█	
Neither agree nor disagree	(3)	0	0.00%		
Agree somewhat	(4)	3	23.08%	██	
Agree strongly	(5)	9	69.23%	█████	
N/A	(0)	0	0.00%		
				0 25 50 100	Question
Response Rate		Mean		STD	Median
13/15 (86.67%)		4.54		0.88	5.00

7 - Compared to other Hopkins courses at this level, the workload for this course is:					
Response Option	Weight	Frequency	Percent	Percent Responses	Means
Much lighter	(1)	0	0.00%		<p>4.58</p>
Somewhat lighter	(2)	0	0.00%		
Typical	(3)	0	0.00%		
Somewhat heavier	(4)	5	38.46%	████	
Much heavier	(5)	7	53.85%	█████	
N/A	(0)	1	7.69%	█	
				0 25 50 100	Question
Response Rate		Mean		STD	Median
13/15 (86.67%)		4.58		0.51	5.00

Course: EN.601.617.01.FA19: Distributed Systems

Instructor: Yair Amir *

Response Rate: 13/15 (86.67 %)

8 - What are the best aspects of this course?

Response Rate	10/15 (66.67%)
----------------------	----------------

- Instructor's teaching ability and deep experience in building distributed systems
- - The course is taught at a very high quality, with great emphasis on system building from scratch. Rarely Hopkins CS courses focus on system building. - The assignments are very challenging, but help you understand the concepts at a very detailed level, and thus enhance your learning. - The theoretical assignments, though initially sound like absurd, are actually very effective, and give you a different outlook to the protocols being studied. - The feedbacks on the assignments are very helpful and very detailed, and queries in the course are handled effectively. - The professor holds design meetings with every group, at least once, before every assignment and project. This helps in going in the right directions, and getting very helpful feedback on your design. - Have never seen a professor work so much with the students, and being so dedicated to the course. The amount of time spent by Yair on the course is insane.
- Yair Amir. I am a fan now. He is the most amazing prof at Hopkins. He has a genuine passion for the subject and I want to be able to inspire people the way he inspires everyone. P.S: Please don't change the course structure
- Instructor has real world experience in building and maintaining distributed systems and covers these experiences in conjunction with theory. Instructor is extremely passionate about teaching as well as his field of study. Instructor and the TA were extremely accessible through piazza and office hours. Assignments were well thought out/designed, and the accessibility of the instructor/TA meant that any vagueness can be easily addressed. Assignments were run on the ugrad machines simulating a distributed network which I thought was cool
- I think the topics are really interesting and the theoretical homework are good challenges
- Very personal sessions with the professor made the feedback all that more meaningful and impactful. The course work is very challenging but I've learned so much, not just on distributed systems but also on system designs in general. He really challenges you to think and gives you so much feedback, it is unlike any other course I have taken.
- Curriculum structure. Projects are just hard enough to require deep thinking and attention to design, but never overwhelming in the extent that by carefully considering what needs to be done and making progress on a consistent basis, the projects can be completed well. Project 2 and the Final Project in particular after completing were two projects where students can be proud. Assignments are not busy-work and if you care about learning the material to its root cause you will be a better system designer by the end of the course. The teaching staff was always available for help and guidance no matter the time and the course structure was well thought out in every lecture and for every project, where everything builds on what came before.
- The lectures are interesting. The instructor teaches passionately. Some lectures are presented interactively in form of a game or something which makes them really interesting. The instructor and TA cared a lot about everyone doing the projects and they put a lot of time for each group (in design meetings) to make projects better.
- The instructor's knowledge, enthusiasm, innovative teaching methods and significant investment in every student. The material was often presented using logic puzzles and was mentally stimulating.
- the projects did good job of making you consider the topic at hand fully

9 - What are the worst aspects of this course?

Response Rate	9/15 (60%)
----------------------	------------

- - The time period of 1.5 month, with all practical and theoretical assignments, can be very stressful, as they are very time consuming. There comes a time when you are not capable of spending more time on the assignments, due to demands of other courses. - Cannot take this course and do amazing in it, if you have even one other medium difficulty course in the semester.
- Nothing!
- Assignments were needlessly time consuming. The assignments were geared towards implementing designs for building robust systems but most of the time was spent debugging mundane code and errors (writing our own data structures and the errors from edge cases).
- The projects are hugs
- The only thing is that in his passion for teaching he often goes over the class time.
- It is a bit of a jump in difficulty in the first project. Many students probably will not have programmed much in C (outside of Intermediate Programming). Getting back into the "trenches" with dynamic memory allocation, etc. was a challenge at first. Additionally, for students who don't have experience with Networking, some of the concepts around TCP/UDP/etc. could have been helped from there being more of a refresher in the first week (i.e. maybe a small conceptual assignment on the basics of networking) before starting Project 1.
- Too much implementation stuff. Although projects are interesting, sometimes I felt like if we had more design-only projects it would have been more helpful. The projects are interesting and fun to implement but there are just too many of them.
- The heavy workload.
- the course seemed dis-organized and both lessons and topics seemed to jump back and forth too much

Course: EN.601.617.01.FA19: Distributed Systems
Instructor: Yair Amir *
Response Rate: 13/15 (86.67 %)

10 - What would most improve this class?

Response Rate 8/15 (53.33%)

- - Having more time for practical assignment 2, would greatly help. - Performance fluctuations when testing your submissions on the ugrad lab, can be very frustrating, when all groups are doing the same close to deadline. A better infrastructure is needed for isolation.
- Nothing! But I would recommend also having a 1 credit course either run in parallel with the course or the semester after the course where we focus on the tools available in the market in addition to spread.
- If the point of the assignments are for students to implement robust designs, having assignments in mixed C/C++ or providing data structure templates in C would greatly speed up the assignment. If less time is spent on debugging, it might be possible to implement multiple designs and compare how they measure against each other.
- If the projects could be somehow bounded. This class requires a lot of homework time
- As mentioned above, a bit more of a refresher on Networking concepts before starting the First Project. Additionally, the Final Project was incredibly deep in scope, which was in my view a welcome challenge. However, the number of edge cases students need to guard against when designing the system (i.e. failures/behavior under all kinds of conditions) could in a sense be never-ending in the context of a school project. Specifically, there were many optimizations we would have liked to make that could have been made to deliver a better project but where we had to prioritize not implementing for the sake of time, and since we get one month for the project with other course works and Thanksgiving, it would be hard to address all those optimizations. Perhaps a discussion in the project specs about the exact things necessary to implement a good solution and a separate discussion of the optimizations that we should consider implementing given more time would have helped more. That way, students wouldn't need to decide for themselves which optimizations are worth spending time implementing over others. But in a sense, this is nitpicking and the class was a wonderful learning experience that I am happy I had the opportunity to take.
- More examples on the presented algorithms. More informative animated presentation slides.
- (1) A reduction in the number of projects, allowing more time per project. (2) More time for drafting the initial design document. (A time period of two days - as was the case for most projects - was not sufficient.)
- planning and origination so that a topic could be fully covered sequentially and a lesson could focus on one area

11 - What should prospective students know about this course before enrolling? (You may comment on any aspect of this course such as assumed background, readings, grading systems, and so on.)

Response Rate 8/15 (53.33%)

- Ability to program in C and a bit knowledge about computer networks.
- - The lectures are intense, and packed with tons of useful and critically important information. You cannot afford to miss the class. - This is the highest quality course currently in the CS department, by far. The learning, and teaching effectiveness, is insane. - DO NOT take this course if you have other hard/medium courses that would take significant time. You would not be able to cope well, and your partner will suffer greatly. - You must be ready to get your hands dirty with C, and dig deep to debug your code.
- Need to know C. I wasn't comfortable with it and it requires some effort to learn but if you are interested, don't let your lack of C skills stop you from taking the course. With that said, don't expect an A
- Data structures as a prerequisite is sufficient. After taking the class, I believe that distributed systems is a field in CS that is heavily underrated
- This class has a big load of work, better to consider how many courses they are taking at the same time of this class. Programming in C is important
- Prof. Amir is perhaps the most dedicated professor I've had at Hopkins when it comes to student outcome having taken two of his classes. I fully got the impression that he would make whatever time necessary to answer student questions (even well after class into the evenings). The TA was also incredibly helpful and would answer questions/help debug even outside office hours. The focus on design was never more apparent here. The projects aren't ones where students can just code without thinking. This approach will lead to redundant work that will need to be scrapped. Students should instead take a few days to consider from end-to-end what they are implementing before writing any significant part of the implementation. If a student wishes to become a better system designer, this should be a class they take. The workload can however be a bit uneven, which students should know about. Project deadlines require several days of dedicated effort, but the times outside project are fairly relaxed. Student success is then contingent on planning well and working even through those relaxed segments in order to not be overwhelmed at the end of deadlines.
- You need to put a lot of time in the projects and assignments. But it sure is an interesting course.
- This course requires: (1) A solid C programming background (2) Significant time investment for the projects (both design and implementation)