Lahore University of Management Sciences

CS 582 – Distributed Systems

Fall 2023

Instructor	Zafar Ayyub Qazi
Room No.	SBASSE 9-G24A
Class timings	
Zoom meeting link for class lectures	https://lums-edu-pk.zoom.us/j/96465611247
Email	zafar.qazi@lums.edu.pk
Instructor Office hours	TBA
TA	TBD
TA Office Hours	TBD
Course URL (if any)	http://lms.lums.edu.pk

Course Teaching Methodology

- Live lectures twice a week during class timings on Zoom
- Lectures will be recorded, and recordings shared with students after each class
- We will use CampusWire for course-related discussions; you can post questions related to lectures on CampusWire. Live questions will be taken over Zoom during the lectures
- CampusWire signup link for the course: https://campuswire.com/p/GD9994F0F
- We will use Poll Everywhere for in class activity.
- The instructor and the TA will be holding online office hours per week via zoom
    - We will use Calendly to schedule meetings with the instructor during office hours.

Course Basics				
Credit Hours	3			
Lecture(s)	Nbr of Lec(s) Per Week	2	Duration	75 minutes
Recitation/Lab (per week)	Nbr of Lec(s) Per Week		Duration	
Tutorial (per week)	Nbr of Lec(s) Per Week		Duration	

Course Distribution	
Core	
Elective	Yes
Open for Student Category	All
Close for Student Category	None

COURSE DESCRIPTION

The goal of this course is to help students gain an understanding of the principles and techniques behind the design of modern, reliable, and high-performance distributed systems.

The course covers fundamentals of distributed systems including, time synchronization, logical clocks, leader elections, consistency models, different algorithms for achieving consensus, failure detection techniques, and techniques for handling failures. Through a variety of case studies, students will also learn how these concepts are applied in various large-scale distributed systems such as Amazon's distributed storage system, Facebook's caching system, Google's global storage system, Big data processing frameworks such as Spark and Distributed Parameter Server, and Blockchains like BitCoin.

Lahore University of Management Sciences

Through this course, students will gain practical experience designing, implementing, and debugging distributed systems through a set of programming assignments. The assignments will be in *GO programming language.*

COURSE PREREQUISITE(S)	
	• Network-Centric Computing (CS 382) or Computer Networks (CS 471) or Grad Standing

COURSE OBJECTIVES	
	• To help students understand the fundamental principles and techniques behind the design of distributed systems. • To empower students to apply these principles and techniques in designing and implementing real-world distributed systems.

Course Learning Outcomes (CLOs)	
	At completion of the course students should be able to:
CLO1	Understand different distributed system models
CLO2	Understand and apply techniques like RPCs for communication in distributed systems
CLO3	Understand, apply, analyze and evaluate time synchronization algorithms (Cristian's algorithm, Berkeley algorithm,
CLO4	NTP)
CLO5	Understand, apply, analyze, and evaluate logical clocks for ordering events and synchronization
CLO6	Understand, apply, analyze, and evaluate election algorithms
CLO7	Understand, apply, analyze, and evaluate fault detectors
CLO8	Understand the tradeoff between strong consistency & availability
CLO9	Understand distributed system properties like *liveness and safety*
CLO10	Understand different consistency models, and determine the suitable consistency model for a given application
CLO11	Understand, apply, and analyze 2-PC
CLO12	Understand, apply, analyze and evaluate consensus algorithms (including, Paxos, Raft, PBFT, Blockchain)
CLO13	Understand, apply, and analyze fault tolerance mechanisms used by Spark to achieve fault tolerance without
CLO14	replication
CLO15	Understand, apply and analyze consistent hashing
CLO16	Understand, apply, and analyze Chord
CLO17	Understand, apply, and analyze ideas presented in Dynamo for designing a highly available key-value store.
CLO18	Understand, apply, and analyze ideas presented in Memcached for designing cache consistency protocols.
CLO19	Understand, apply, and analyze ideas presented in Zookeeper for designing a replicated distributed coordination
CLO20	service. Understand distributed transactions and ACID properties
CLO21	Understand, apply, and analyze ideas presented in Spanner for designing a globally distributed data store.
CLO22	Understand, apply, and analyze the ideas presented Distributed Parameter Server for designing scalable ML
	frameworks.
	Design, implement and debug distributed system components
	Understand and analyze the design of a distributed systems by reading its technical description from a published paper

Lahore University of Management Sciences

CLO	CLO Statement	Bloom's Cognitive Level	PLOs/Graduate Attributes (Seoul Accord)
CLO1	Understand different distributed system models	C2	PLO2
CLO2	Understand and apply techniques like RPCs for communication in distributed systems	C2, C3	PLO2, PLO3
CLO3	Understand, apply, analyze and evaluate time synchronization algorithms (Cristian's algorithm, Berkeley algorithm, NTP)	C2, C3, C4	PLO2, PLO3, PLO5
CLO4	Understand, apply, analyze, and evaluate logical clocks for ordering events and synchronization	C2, C3, C4	PLO2, PLO3, PLO4, PLO5
CLO5	Understand state of a distributed system, and Chandy-Lamport algorithm for checkpointing the state of distributed system	C2	PLO2
CLO6	Understand, apply, analyze, and evaluate election algorithms	C2, C3, C4	PLO2, PLO3, PLO4, PLO5
CLO7	Understand, apply, analyze, and evaluate fault detectors	C2, C3, C4	PLO2, PLO3, PLO4, PLO5
CLO8	Understand the tradeoff between strong consistency & availability	C2	PlO2
CLO9	Understand distributed system properties like liveness and safety	C2	PLO2
CLO10	Understand, apply, analyze and evaluate consensus algorithms (including, Paxos, Raft, PBFT, Blockchain)	C2, C3, C4	PLO2, PLO3, PLO4, PLO5
CLO11	Understand, apply, analyze and evaluate consensus algorithms (including, Paxos, Raft, PBFT, Blockchain)	C2, C3, C4	PLO2, PLO3, PLO4, PLO5
CLO12	Understand and analyze fault tolerance mechanisms used by Spark to achieve fault tolerance without replication	C2, C4	PLO2, PLO3
CLO13	Understand and analyze DHTs, and consistency	C2, C4	PLO2, PLO3
CLO14	Understand and analyze Chord	C2, C4	PLO2, PLO3
CLO15	Understand, analyze and evaluate the design of Dynamo	C2, C4, C6	PLO2, PLO4, PLO5
CLO16	Understand, analyze and evaluate the design of Memcached	C2, C4, C6	PLO2, PLO4, PLO5
CLO17	Understand distributed transactions and ACID properties	C2	PLO2
CLO18	Understand, analyze and evaluate the design of Spanner	C2, C4, C6	PLO2, PLO4, PLO5
CLO19	Understand, analyze and evaluate the design of Distributed Parameter Server Design, implement and debug distributed system components	C2, C4, C6	PLO2, PLO3, PLO4, PLO5
CLO20	Understand and analyze the design of a distributed systems by reading its technical description from a published paper	C2, C4	PLO2, PLO3, PLO4, PLO5

Grading Breakup and Policy

Programming Assignment(s): 35%
Quizzes: 25% (N-2 policy) (in-class, announced)
Homeworks: 15% (related to assigned paper reading, due at 11:59pm the day before the class)
Final examination (comprehensive): 25% (synchronous exam)

This fall, CS 582 will be run entirely online. Below, we describe how each type of assessment will be conducted.

Assessment	Weight (%)	Related CLOs

Quizzes (35%)		
Paper summaries:		
Final Examination		

Quizzes

- Quizzes will be announced, in-class, and follow N-2 policy.
- We will have a total of 9 quizzes in the course but only your best 7 quizzes will count towards your grade. An important purpose of having N-2 quizzes is to account for all issues (e.g., Internet connectivity, electricity outage, sickness, etc.) that may prevent you from taking the quiz. No request for a makeup quiz will be entertained if you miss up to 2 quizzes irrespective of the reason. In the exceptional situation in which you end up missing more than 2 quizzes, we will consider a makeup quiz only if there is a valid justification. In that case, we reserve the right to determine the mode of the quiz, which may be oral or textual.
- All quizzes will take place during class timings and will be 10-15mins in duration.
- All quizzes will be announced (see the schedule below for the dates).
- The syllabus for every quiz will include lecture in which the quiz will be taken.

Exam

- There will be a comprehensive final exam, which will be conducted synchronously. There will be no midterm exam in the course.
- The final exam will take place during the final exam week.

Programming Assignments

- Programming assignments are an integral part of this course, and intended to provide students practical experience designing, implementing and debugging distributed systems. The assignments will be in GO language. There will be a total of four programming assignments spread over the semester.
- The first two assignments must be done individually, whereas the last two assignments can be done in pairs.
- Please note the following policies regarding the programming assignments. You are responsible adhering to these policies.
    - All deadlines are hard
    - Re-grading can be requested within 2 days after grade reporting
    - Students must not share actual program code with other students.
    - Student must never have someone else's code/assignment in your possession at any time
    - Students must be prepared to explain any program code they submit.
    - Students must indicate with their submission any assistance received.
    - All submissions are subject to plagiarism detection.
    - Students cannot copy code from the Internet.
    - Students are strongly advised that any act of plagiarism will be reported to the Disciplinary Committee
    - If in doubt about any policy, talk to the course staff
- *Late Submission:* You should submit your work on an assignment on LMS before its due time. The assignments will generally be divided into two parts, with split deadlines. If you submit your work late for any assignment (part), we will award you a fraction of the score you would have earned on the assignments had it been turned in on time, according to this sliding scale:
    - 90% for work submitted up to 24 hours late
    - 80% for work submitted up to 2 days late
    - 70% for work submitted up to 3 days late
    - 60% for work submitted up to 5 days late
    - 50% for work submitted after 5 days late

For example, if you should have earned 8/10 points but submitted 36 hours late, you will instead earn 6.4 points.

Besides the above policy, you are allowed **five "free" late days** during the semester (that can be applied to one of assignments 1 through 3; the final assignment will be due tentatively on final day of classes **[14 Dec]** and cannot be turned late. ***The last day to do any late submission is also the final day of classes, even if you have free late days remaining*).** You do not need to tell us that you are applying your "late day" -- we'll remove the late penalty at the end of the semester from the assignment that benefits you the most.

Homeworks

- Related to assigned paper reading, due at 11:59pm the day before the class. Only for selected lectures (with mandatory paper reading), highlighted below in the class schedule.
- For each paper, we will have specific questions related to the paper that you will have to answer in your homework.

Academic Honesty

The principles of truth and honesty are recognized as fundamental to a community of teachers and students. This means that all academic work will be done by the student to whom it is assigned without unauthorized aid of any kind. Plagiarism, cheating and other forms of academic dishonesty are prohibited. Any instances of academic dishonesty in this course (intentional or unintentional) will be dealt with swiftly and severely. Potential penalties include receiving a failing grade on the assignment in question or in the course overall. For further information, students should make themselves familiar with the relevant section of the LUMS student handbook.

We believe that most students can distinguish between helping other students understand course material and cheating. Explaining a subtle point from lecture or discussing course topics is an interaction that we encourage. However, all coursework must be completed individually and independently unless explicitly stated otherwise (e.g., in case of pair assignments). We have various methods of detecting cheating– so don't do it! We also ask that you do not post your assignment solutions publicly.

Cheating besides being unethical also has many profound negative consequences:

- Takes away your opportunity for learning and lowers your confidence
- You'd never get this time back!
- Negatively impacts your colleagues
- We all are surely better than that!

Rather than copying someone else's work, ask for help. You are not alone in this course! The entire course staff is here to help you succeed. If you invest the time to learn the material and complete the assignments, you won't need to copy any answers

We want you to succeed!

If you are feeling overwhelmed, come to our office hours and talk with us. We know university life can be stressful – and especially so during the COVID-19 pandemic – and we want to help you succeed.

Harassment Policy

SSE, LUMS and particularly this class, is a harassment free zone. There is absolutely zero tolerance for any behaviour that is intended or has the expected result of making anyone uncomfortable and negatively impacts the class environment, or any individual's ability to work to the best of their potential. In case a differently-abled student requires accommodations for fully participating in the course, students are advised to contact the instructor so that they can be facilitated accordingly.

If you think that you may be a victim of harassment, or if you have observed any harassment occurring in the purview of this class, please reach out and speak to me. If you are a victim, I strongly encourage you to reach out to the Office of Accessibility and Inclusion at oai@lums.edu.pk or the sexual harassment inquiry committee at harassment@lums.edu.pk for any queries, clarifications, or advice. You may choose to file an informal or a formal complaint to put an end of offending behavior. You can find more details regarding the LUMS sexual harassment policy here. To file a complaint, please write to harassment@lums.edu.pk.

SSE Council on Equity and Belonging

Lahore University of Management Sciences

In addition to LUMS resources, SSE's Council on Belonging and Equity is committed to devising ways to provide a safe, inclusive and respectful learning environment for students, faculty and staff. To seek counsel related to any issues, please feel free to approach either a member of the council or email at cbe.sse@lums.edu.pk
Rights and Code of Conduct for Online Teaching
A misuse of online modes of communication is unacceptable. TAs and Faculty will seek consent before the recording of live online lectures or tutorials. Please ensure if you do not wish to be recorded during a session to inform the faculty member. Please also ensure that you prioritize formal means of communication (email, LMS) over informal means to communicate with course staff.

Makeup Policy

- Please refer to Student Handbook 2019-20, page 37, article 25, titled "Makeup Policy for Graded Instruments".
- *"In case N-X policy is implemented for an instrument having multiple sub instruments then petitions will not be accepted for that instrument".*

Examination Detail	
Midterm Exam	Yes/No: No
Final Exam	Yes/No: Yes Combine Separate: Duration: Exam Specifications:

Code of Conduct

1. When attending classes, please ensure that your video is turned off and your mic is muted unless you are asked to do so.
2. **Only authenticated users will be to join class lectures on zoom – please make you join the zoom lectures through an account based on your LUMS email address.**
3. All quizzes will be announced (in fact, they have already been specified in the schedule below) and students must ensure that their devices are charged, and they have a stable internet connection (including smartphones).
4. All assessments including quizzes and the final exam will be timed. Make sure that you are able to start them on time.

Schedule

Lecture	Topics	Readings	Assessments/ Objectives	Related CLOs
1	Course Overview and Introduction • What are distributed systems? • Why study them? • Course structure, grading, workload and staff	Chapter 1		
Fundamentals: (1) Communication, (2) Coordination (3) Group membership & Failure detection, and (4) Replication and Consistency				
2	System Models and Network Communication • System models • Overview of layering and communication fundamentals	Chapter 4.1 Refer to slides	CLO1	

Tutorial#1	Introduction to GO	GO Resources: • A Tour of Go • Effective Go • Introduction to Asynchronous programming • Profiling Go Programs • Checking Test coverage in Go		
3	Remote Procedure Calls • Why RPCs? • How do they work? • RPC Issues	Chapter 4.2	CLO2	
Tutorial#2	RPCs in GO			
4	Time Synchronization • "Wall clock time" synchronization • Cristian's algorithm, Berkeley algorithm	Chapter 6.1 and 6.2	CLO3	
5	Time Synchronization (Cont'd) • NTP	Chapter 6.2	**Quiz 1** CLO3	
6	Lamport Clocks • Lamport Clocks • Totally ordered Multicast using Lamport Clocks	Chapter 6.2	CLO4	
7	Vector Clocks • Vector Clocks • Bulletin board example	Chapter 6.2 Refer to slides	CLO4	
8	Leader Election Algorithms • Bully algorithm • Ring algorithm	Chapter 6.4	**Quiz 3** CLO5	
9	Leader election (cont'd) • Leader Election in Raft	Refer to slides	CLO5	
10	Failure Detectors • Important properties of failure detectors • Failure detection in synchronous and asynchronous systems • Different types of failure detectors	Refer to slides	CLO6	
11	Replication and Consistency • CAP Theorem • Consistency models: linearizability, sequential consistency, eventual consistency, etc.	Chapter 7.1, 7.2, 7.3 & 7.5	CLO7-9 **Quiz 4**	
Fault Tolerance: (1) 2-PC, (2) Paxos, (3) Raft, (4) PBFT, (5) Blockchains, (5) Spark				

12	Introduction to fault tolerance (Consistency Protocols and Consensus) • Type of consistency protocols • Two Phase Commit (2-PC)	Chapter 8.1 and 8.2	CLO10	
13	Paxos • Consensus Problem • Paxos Algorithm	Chapter 8.2 Paxos Made Simple (mandatory reading before the class)	CLO11, CLO22	
14	Fault Tolerance with Raft	In Search of an Understandable Consensus Algorithm by Diego et al. (mandatory reading before the class)	CLO11, CLO22	
15	Fault Tolerance with Raft (cont'd), Byzantine fault tolerance	Chapter 8.2	**Quiz 5** CLO11	
16	Practical Byzantine Fault Tolerance	Chapter 8.2 Practical Byzantine Fault Tolerance by Barbara Liskov et al. (mandatory reading before the class)	CLO 11, CLO22	
17	Decentralized consensus and Blockchain	Bitcoin: A Peer-to-Peer Electronic Cash System (mandatory reading before the class)	CLO 11, CLO22	
18	Decentralized consensus and Blockchain (cont'd)		CLO 11	
19	Big Data and Spark	Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing by Matai et al. (mandatory reading before the class)	CLO 12, CLO22 **Quiz 6**	
Distributed Data Stores and Services: (1) P2P storage, (2) Highly available key-value stores, (3) Cache consistency, (4) Globally consistent data stores, (5) Distributed ML framework				
20	Data partitioning and lookups	Refer to slides	CLO 13	
21	A scalable peer-to-peer service: Chord	Chord by Ion Stoica et al. (mandatory reading before the class)	**Quiz 7** CLO14, CLO22	
22	Highly Available Key-value Stores: Amazon's Dynamo	Dynamo: Amazon's Highly Available Key-value Store by Giuseppe et al. (mandatory reading before the class)	CLO15, CLO22	
23	Cache consistency: Memcached at Facebook	Scaling Memcache at Facebook by Rajesh et al. (mandatory reading before the class)	CLO16, CLO22	

24	Wait-free coordination for Internet-scale systems: Zookeeper	Zookeeper by Patrick et al. (mandatory reading before the class)	CLO17, CLO22	
25	Distributed Transactions and Concurrency	Refer to slides	CLO18, CLO22 **Quiz 8**	
26	Google's Globally Distributed Database: Spanner	Spanner by James et al. (mandatory reading before the class)	CLO19, CLO22	
27	Spanner (Cont'd)		CLO19, CLO22 **Quiz 9**	
28	Distributed Parameter Server for Machine Learning	Scaling machine learning with Parameter Server by Mu Li et al. (mandatory reading before the class)	CLO20, CLO22	

Textbook(s)/Supplementary Readings
Required Text • Distributed Systems: Principles and Paradigms, 3rd Edition, by Andrew S. Tanenbaum and Maarten Van Steen **Optional Texts** • Distributed Systems: Concepts and Design – 4th Ed., George Colouris, Jean Dollimore, Tim Kindberg. Pearson 2006 • Principle of Distributed Computing, Lecture Notes by Roger Wattenhofer, ETH Zurich • Distributed Systems for fun and profit by Mikito Takada [Elevate this and consider adding chapters from here] • Other supplementary readings will include research papers from top tier networking conferences such as SOSP, OSDI, SIGCOMM, NSDI

Please note this course outline can be subject to some changes.

Lahore University of Management Sciences

Appendix A
Bloom's Taxonomy

BLOOM's TAXONOMY*	
1 - Remember 2 - Understand 3 - Apply 4 - Analyze 5 - Evaluate 6 - Create	• Recall facts and basic concepts • Explain ideas or concepts • Use information in new situations • Draw connection among ideas • Justify a stand or decision • Produce new or original work

https://cft.vanderbilt.edu/guides-sub-pages/blooms-taxonomy/