# INSTITUTE OF AERONAUTICAL ENGINEERING

**(AUTONOMOUS)**

Dundigal, Hyderabad - 500 043

## COMPUTER SCIENCE AND ENGINEERING

## COURSE DESCRIPTION FORM

| | | | | |
|---|---|---|---|---|
| **Course Title** | **DISTRIBUTED SYSTEMS** | | | |
| **Course Code** | **A60521** | | | |
| **Regulation** | R15 - JNTUH | | | |
| **Academic Year** | 2017-2018 | | | |
| **Course Structure** | Lectures | Tutorials | Practicals | Credits |
| | 4 | - | - | 4 |
| **Course Coordinator** | Mr. RM Noorullah, Associate Professor, CSE<br>Mr. N V Krishna Rao , Associate Professor, CSE | | | |
| **Team of Instructors** | Mr. Ch Srikanth, Assistant Professor, CSE<br>Mr. Rakesh, Assistant Professor, CSE. | | | |

## I. COURSE OVERVIEW:

The primary objective of this course is to introduce the concept of theoretical, technical and programming aspects of advanced distributed systems and computing and the core elements of such advanced systems. The operation, characteristics and of major computer networks are studied because of their strong influence on programming interfaces (APIs) and application design. Topics include multithreading, network programming, consistency, fault tolerance, consensus, security, and several case studies of distributed systems. Finally, this course also covers issues and solutions related to the design and the implementation of distributed applications using socket programming

## II. COURSE PREQUISITES:

| Level | Course Code | Semester | Prerequisites | Credits |
|---|---|---|---|---|
| UG | A50515 | I | Computer Networks | 4 |
| UG | A60522 | II | Information security | 4 |
| UG | ACS507 | II | Network Programming and Management | 4 |

## III. MARKS DISTRIBUTION:

| Sessional Marks | University End Exam marks | Total marks |
|---|---|---|
| There shall be 2 midterm examinations. Each midterm Examination consists of subjective type and Objective type tests.<br>The subjective test is for 10 marks, with duration of 1 hour. Subjective test of each semester shall contain 4 questions, the student has to answer 2 questions, each carrying 5 marks.<br><br>Objective type test is for 10 marks with duration of 20 minutes. It consists of 10 multiple choice and 10 objective type | 75 | 100 |

| Sessional Marks | University End Exam marks | Total marks |
|---|---|---|
| questions, the student has to answer all the questions and each carries half mark.<br><br>First midterm exam shall be conducted for the first two and half units of syllabus and second midterm examination shall be conducted for the remaining portion.<br>Five marks are marked for assignments. There shall be two assignments in every theory course. Marks shall be awarded considering the average of two assignments in each course. | | |

## IV. EVALUATION SCHEME:

| S. No | Component | Duration | Marks |
|---|---|---|---|
| 1. | I Mid Examination | 80 minutes | 20 |
| 2. | I Assignment | - | 5 |
| 3. | II Mid Examination | 80 minutes | 20 |
| 4. | II Assignment | - | 5 |
| 5. | External Examination | 3 hours | 75 |

## V. COURSE OBJECTIVES:

**The course should enable the students to:**

I. Understand basic concepts of a distributed system and sharing of resources in distributed manner.
II. Describe the theoretical concepts, namely, virtual time, agreement and consensus protocols.
III. Demonstrate the concepts of IPC, Group communication & RPC.
IV. Illustrating the methods of the DFS and DSM concepts.
V. Understand the concepts of transaction in distributed environment and associated concept, namely, concurrency control, deadlocks and error recovery.

## VI. COURSE OUTCOMES:

**Students, who complete the course, will have demonstrated the ability to do the following:**

1 Implement and describe the structure of distributed systems programs.
2 Write programs that can interoperate using well-defined protocols and debug the code on multiple machines of multiple cores.
3 Understand about distributed algorithms for locking, synchronization and concurrency, scheduling, and replication and use standard primitives such as UDP and TCP.
4 Understand the general properties of networked communication necessary for distributed systems programming in clusters and on the Internet.
5 Employ and create common paradigms for easing the task of distributed systems programming and able to clearly elucidate their benefits, drawbacks, and limitations.

6 Identify the security challenges faced by distributed systems programs and able to select appropriate security solutions to meet the needs of commonly encountered distributed programming scenarios.

7 Comprehend and design a new distributed system with the desired features and able to develop new distributed applications.

## VII. HOW PROGRAM OUTCOMES ARE ASSESSED:

| Program Outcomes | | Level | Proficiency assessed by |
|---|---|---|---|
| PO1 | **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems. | H | Assignments and Tutorials |
| PO2 | **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences. | N | ---- |
| PO3 | **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations. | H | Assignments and Tutorials |
| PO4 | **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions. | S | Open ended experiments / |
| PO5 | **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations. | H | Tutorials |
| PO6 | **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice. | N | ---- |
| PO7 | **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development. | N | ---- |
| PO8 | **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. | N | ---- |
| PO9 | **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings. | S | Certifications |
| PO10 | **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions. | N | ---- |

| Program Outcomes | | Level | Proficiency assessed by |
|---|---|---|---|
| PO11 | **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments. | N | ---- |
| PO12 | **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change. | H | Assignments |

**N - None S - Supportive H - Highly Related**

## VIII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

| Program Specific Outcomes | | Level | Proficiency assessed by |
|---|---|---|---|
| PSO1 | **Professional Skills:** The ability to research, understand and implement computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient analysis and design of computer-based systems of varying complexity. | H | Lectures, Assignments |
| PSO2 | **Problem-Solving Skills:** The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success. | S | Projects |
| PSO3 | **Successful Career and Entrepreneurship:** The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies. | H | Guest Lectures |

**N - None S - Supportive H- Highly Related**

## IX. SYLLABUS:

| **UNIT – I** |
|---|
| **Characterization of Distributed Systems:** Introduction, Examples of Distributed Systems, Resources Sharing and the Web, Challenges.<br>**System Model:** Introduction, Architectural Models, Fundamental Models. |
| **UNIT – II** |
| **Time and Global States:** Introduction, clocks Events and process states, synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging.<br>**Coordination and agreement:** Introduction, Distributed Mutual Exclusion, Elections, Multicast Communications, Consensus and Related Problems. |
| **UNIT – III** |
| **Inter Process Communication:** Introduction, The API for the internet Protocols, External Data Representation and Marshalling, Client-Server Communication, Group Communication, Case Study: IPC in UNIX.<br>**Distributed Objects and Remote Invocation:** Introduction, communication between distributed objects, remote procedure call, Events and Notifications, case study: JAVA RMI. |

| UNIT-IV |
|---|
| **Distributed file system:** Introduction, file service architecture, case study1:sun network file system, case study2:the Andrew file system.<br>**Name services:** Introduction, Name service and the domain name system, directory services, case study of the global name services.<br>**Distributed shared memory:** Introduction, design and implementation issues, sequential consistency and IVY case study release consistency, MUNIN case study, and other consistency models. |
| **UNIT –V** |
| **Transactions and Concurrency Control:** Introduction, Transactions, Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control.<br>**Distributed Transactions**: Introduction, Flat and Nested Distributed transactions, Atomic Commit Protocols, Concurrency Control in Distributed transactions, distributed deadlocks, Transaction Recovery |

**TEXT BOOKS:**

| | |
|---|---|
| **1** | George Coulouris, J Dollimore and Tim Kindberg, "Distributed System, Concepts and Design", Pearson Education, 4th Edition, 2009. |

**REFERENCES:**

| | |
|---|---|
| **1** | Andrew S. Tanenbaum, Maarten Van Steen, "Distributed Systems, Principles and Paradigms", 2nd Edition, PHI 2007. |
| **2** | Sukumar Ghosh, Chapaman & Hall/ CRC, Taylor & Fransis Group, "Distributed Systems, An Algorithm Approach", 2007. |

## X. COURSE PLAN:

The course plan is meant as a guideline. There may probably be changes.

| Lecture No | Topics to be covered | Course Learning Outcomes | Reference |
|---|---|---|---|
| 1-5 | Characterization of Distributed Systems**:** Introduction, Examples of Distributed Systems, Resources Sharing and the Web, Challenges. | **Explain** the importance of Distributed Systems Distributed Systems distributed | T1:1-7 |
| | | **Demonstrate** the challenges in the distributed systems | T1:8-16 |
| 6-8 | System Model: Introduction, Architectural Models. | **Illustrate** the models in distributed systems with their examples. | T1:29-30 |
| 9-10 | Fundamental Models. | **Demonstrate** different fundamental models | T1:31-47 |
| 11-15 | Time and Global States: Introduction, clocks Events and process states, | **Explain** event and process States. | T1:385-387 |
| 16-17 | Global states, distributed debugging | **Define** the difference between the physical clocks and logical clocks | T1:389-397 |
| | | **Explain** the debugging process in the distributed systems | T1:400-409 |
| | | **Explain** the mutual exclusion | T1:419-423 |
| 18-20 | Coordination and agreement: Introduction, Distributed Mutual Exclusion, Elections, Multicast Communications, Consensus and | **Explain** the election procedure and their algorithms | T1:436-462 |
| | | **Define** IPC. | T1:125-127 |

| | | | |
|---|---|---|---|
| | Related Problems. | | |
| 21-24 | Inter Process Communication: Introduction, The API for the internet Protocols, External Data Representation and Marshalling. | **Define** data marshalling | T1:138-145 |
| | | **Explain** client server communication | T1:146-158 |
| 25-27 | Client-Server Communication, Group Communication, Case Study: IPC in UNIX. | **Explain** RPC and RPI | T1:165-169 |
| 28-31 | Distributed Objects and Remote Invocation: introduction, communication between distributed objects. | **Define** file services and their procedures. | T1:183-323 |
| 32-36 | Remote procedure call, Events and Notifications, case study: JAVA RMI. Distributed file system: Introduction, file service architecture, case study1:sun network file system | **Explain** name services and their instances | T1:335-374 |
| 37-40 | Case study2: the Andrew file system, Name services: Introduction, Name service and the domain name system, directory services, case study of the global name services. | **Define** shared memory in distributed systems | T1:635-640 |
| 41-44 | Distributed shared memory: Introduction, design and implementation | **Explain** types of consistency in distributed systems | T1:649-663 |
| 45-48 | Sequential consistency and IVY case study release consistency, MUNIN case study, and other consistency models. | **Explain** concurrency execution of transactions in the distributed systems | T1:465-480 |
| 49-52 | Transactions and Concurrency Control: Introduction, Transactions, Nested | **Demonstrate** the locks and time stamp ordering methods | T1:482-508 |
| 53-58 | Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control. | **Explain** types of transactions and their properties | T1:515-539 |
| 59-65 | Distributed Transactions: Introduction, Flat and Nested Distributed transactions, Atomic Commit Protocols, Concurrency Control in Distributed transactions, distributed deadlocks, Transaction | | |

## XI. MAPPING COURSE OBJECTIVES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

| Course Objectives | Program Outcomes | | | | | | | | | | | | Program Specific Outcomes | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 | PSO3 |
| I | | S | H | | | | | | | | | | H | S | H |
| II | | H | | | S | | | | | | | | S | | H |
| III | | | | | | | | | | H | S | | H | S | S |
| IV | | | S | | H | | | | | | | | H | S | S |
| V | | | | | | | | | | | S | H | S | | H |

**S – Supportive** **H - Highly Related**

## XII. MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

| Course Outcomes | Program Outcomes | | | | | | | | | | | | Program Specific Outcomes | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 | PSO3 |
| 1 | H | | | | | S | | | | | | | H | | H |
| 2 | | S | | | | | | | | | H | | S | H | |
| 3 | | H | S | | | | | | | | | | | H | H |
| 4 | | | H | | S | | | | | | | | S | H | |
| 5 | | | | | H | | | | | | S | | H | | S |
| 6 | | | | | S | | | H | | | | | H | H | S |
| 7 | | H | | | | | | | | | S | | H | | S |

**S – Supportive** **H - Highly Related**

**Prepared by:** Mr. RM Noorullah, Associate Professor, CSE
Mr. N V Krishna Rao, Associate Professor, CSE
Mr. Ch Srikanth, Assistant Professor, CSE
Mr. M Rakesh Assistant Professor, CSE

**HOD, COMPUTER SCIENCE AND ENGINEERING**