SEMESTER VI

19CSE314	SOFTWARE ENGINEERING	L-T-P-C: 2-0-3-3

Course Objectives

- This course addresses issues in the engineering of software systems and development using live case studies from industries.
- The objectives of this course are to introduce basic software engineering concepts; to introduce the Agile Software development process; hands-on training (experiential learning) using state-of-the-art tools to understand the concepts learnt in the class.
- The course helps students to be industry-ready in terms processes, tools and terminologies from agile and devops point of view

Course Outcomes

CO 1: Understand and apply the principles of software engineering
CO 2: Understand various software process models
CO 3: Apply the appropriate software design methodology for a given scenario
CO 4: Evaluate a system developed for real-world applications in Agile Mode
CO 5: Understand and implement various industry standards

CO-PO Mapping

PO/PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1											1	1
CO2	3											2	3	1
CO3	3	3	3			2	2			2		3	3	3
CO4		2	3	3	3	2	2	3	3	3	2		3	3
CO5					3				3	1	2	3	3	3

Syllabus

Unit 1
Process Models – overview, Introduction to Agile, Agile Manifesto, principles of agile manifesto, over-view of Various Agile methodologies - Scrum, XP, Lean, and Kanban, Agile Requirements - User personas, story mapping, user stories, estimating and prioritizing stories, INVEST, acceptance criteria, Definition of Done, Release planning Key aspects of Scrum: roles - Product Owner, Scrum Master, Team, Manager in scrum and product backlog Scrum process flow: product backlog, sprints backlog, scrum meetings, demos. How sprint works: Sprint Planning, Daily scrum meeting, updating sprint backlog, Burn down chart, sprint review, sprint retrospective. Scrum Metrics-velocity, burn down, defects carried over.

Unit 2
Traditional process Models: Waterfall, incremental, evolutionary, concurrent. Requirements Engineering: Tasks Initiation-Elicitation-Developing Use Cases-Building the analysis Model-Negotiation- Validation Requirements Modelling - building the analysis model, Scenario based methods, UML Models, Data Models. Design engineering

Design concepts, Design models, software architecture, architectural styles and patterns, Architectural design: styles and patterns, architectural design, Refining architecture to components. Performing user interface Design-Golden Rules-User Interface Analysis and Design- Interface Analysis-Interface design steps.

Unit 3

Testing strategies and tactics: Unit testing, integration testing, validation and system testing, Devops.

Text Book(s)

Pressman R S, Bruce R.Maxim, Software Engineering - A Practitioner's Approach. Eighth Edition, McGraw-Hill Education, 2019.

Reference(s)

Crowder JA, Friess S. Agile project management: managing for success. Cham: Springer International Publishing; 2015.
Stellman A, Greene J. Learning agile: Understanding scrum, XP, lean, and kanban. " O'Reilly Media, Inc."; 2015.
Gregory J, Crispin L. More agile testing: learning journeys for the whole team. Addison-Wesley Professional; 2015.
Rubin KS. Essential Scrum: a practical guide to the most popular agile process. Addison-Wesley; 2012.
Cohn M. User stories applied: For agile software development. Addison-Wesley Professional; 2004.

Evaluation Pattern:

Assessment	Internal	External
Periodical 1	10	
Periodical 2	10	
*Continuous Assessment (Theory) (CAT)	10	
Continuous Assessment (Lab) (CAL)	40	
End Semester		30

*CA – Can be Quizzes, Assignment, Projects, and Reports.

19CSE313	PRINCIPLES OF PROGRAMMING LANGUAGES	L-T-P-C: 2-0-3-3

Course Objectives

- This course provides a quick overview of different paradigms of programming languages.
- It focuses primarily on the functional programming paradigm using Haskell and Scala and discusses the concurrent programming paradigm using Java.

Course Outcomes

CO1: Understand and write pure functional programs (especially in Haskell and Scala).
CO2: Understand and write concurrent programs in Java.
CO3: Formulate abstractions with higher order procedures.
CO4: Formulate abstractions with data.

CO-PO Mapping

PO/PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1		3	2	1						3	2
CO2	2	2	2	1	3	2	1						3	2
CO3	3	3	3	3	3	2	2						3	2
CO4	3	3	3	3	3	2	2						3	2

Syllabus

Unit 1
Programming Paradigms – Overview of different programming paradigms. Functional Programming with Haskell – functions and types, functional composition, numbers, lists, tuples, type classes, pattern matching, higher order functions: currying, lambdas, maps and filters, folds, IO monad

Unit 2
Concurrency in Java - Issues with concurrency: safety, liveness, fairness, Threads, locks and synchronization, Thread pools, Futures and callables, fork-join parallel framework

Unit 3
Functional Programming overview with Scala – Basic types and operations, classes and objects, functional objects, functions and closure, composition and inheritance

Text Book(s)
Bird R. Thinking functionally with Haskell. Cambridge University Press; 2014.
Martin Odersky, Lex Spoon and Bill Venners, "Programming in Scala – A Comprehensive Step-by-Step Guide", Third Edition, Artima Inc. 2016.

Reference(s)
Herbert Schildt. Java: The Complete Reference. Eleventh Edition, McGrawHill Education; 2018.

Evaluation Pattern:

Assessment	Internal	External
Periodical 1	10	
Periodical 2	10	
*Continuous Assessment (Theory) (CAT)	10	
Continuous Assessment (Lab) (CAL)	40	
End Semester		30

*CA – Can be Quizzes, Assignment, Projects, and Reports.

19CSE312	DISTRIBUTED SYSTEMS	L-T-P-C: 3-0-3-4

Course Objectives

- This course is an introduction to the design of distributed systems and algorithms that support distributed computing.
- It aims to provide a practical exposure into the design and functioning of existing distributed systems.

Course Outcomes

CO1: Understand the design principles in distributed systems and the architectures for distributed systems.
CO2: Apply various distributed algorithms related to clock synchronization, concurrency control, deadlock detection, load balancing, voting etc.
CO3: Analyze fault tolerance and recovery in distributed systems and algorithms for the same.
CO4: Analyze the design and functioning of existing distributed systems and file systems.
CO5: Design and implement a simple distributed system and implement different distributed algorithms over it.

CO-PO Mapping

PO/PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	2		2							3	2
CO2	3	3	2	2		3							3	2
CO3	3	3	2	2		2	3	2	2			2	3	2
CO4	3	3	2	2	2	3		2					3	2
CO5	3	3	3	1	3	3	1	3	2	2	2	2	3	2

Syllabus

Unit 1
A Taxonomy of Distributed Systems - Models of computation: shared memory and message passing systems, synchronous and asynchronous systems. Communication in Distributed Systems: Remote Procedure Calls, Message Oriented Communications and implementations over a simple distributed system.

Unit 2
Global state and snapshot algorithms. Logical time and event ordering, clock synchronization, Distributed mutual exclusion, Group based Mutual Exclusion, leader election, concurrency control, deadlock detection, termination detection, implementations over a simple distributed system.

Unit 3
Consistency and Replication: Data Centric Consistency, Client Centric Consistency, Replica Management, Consistency Protocols. Fault tolerance and recovery: basic concepts, fault models, agreement problems and its applications, commit protocols, voting protocols, check pointing and recovery. Distributed file systems: scalable performance, load balancing, and availability.
Case Studies: Dropbox, Google FS (GFS)/ Hadoop Distributed FS (HDFS), Bigtable/HBase MapReduce, RDDs, Apache Spark

Text Book(s)
Andrew S. Tannenbaum and Maarten van Steen, Distributed Systems: Principles and Paradigms, Third Edition, Prentice Hall, 2017.

Reference(s)

Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press, 2011.
Garg VK, Garg VK. Elements of distributed computing. John Wiley & Sons; 2002.
George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, Distributed Systems: Concepts and Design, Fifth Edition, Pearson Education, 2017.
Fokkink W. Distributed algorithms: an intuitive approach. Second Edition, MIT Press; 2018.

Evaluation Pattern:

Assessment	Internal	External
Periodical 1	10	
Periodical 2	10	
*Continuous Assessment (Theory) (CAT)	15	
Continuous Assessment (Lab) (CAL)	30	
End Semester		35

*CA – Can be Quizzes, Assignment, Projects, and Reports.

19CSE311	COMPUTER SECURITY	L-T-P-C: 3-0-0-3

Course Objectives

- This course provides basic knowledge and skills in the fundamental theories and practices of cyber security.
- It provides an overview of the field of security and assurance emphasizing the need to protect information being transmitted electronically.

Course Outcomes

CO1: Understand the fundamental concepts of computer security and apply to different components of computing systems.
CO2: Understand basic cryptographic techniques.
CO3: Understand how malicious attacks, threats, security and protocol vulnerabilities impact a system's Infrastructure.
CO4: Demonstrate knowledge in terms of relevance and potential of computer security for a given application.

CO-PO Mapping

PO/PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	1										3	2
CO2	3	3	3	1	2								3	2
CO3	3	3	3	2		2		3					3	2
CO4	3	3	1	2	3	2		2					3	2

Syllabus

Unit 1
Basics of Computer Security: Overview – Definition of terms – Security goals – Shortcomings – Attack and defense – Malicious code – Worms – Intruders – Error detection and correction Encryption and Cryptography: Ciphers and codes – Public key algorithms – Key distribution – Digital signatures.

Unit 2

Security Services: Authentication and Key Exchange Protocols - Access control matrix – User authentication – Directory authentication service – Diffie-Hellman key exchange – Kerberos.

Unit 3
System security and Security models: Disaster recovery - Protection policies. E-mail Security: Pretty good privacy - Database Security: Integrity constraints - Multi-phase commit protocols - Networks Security: Threats in networks - DS authentication -Web and Electronic Commerce: Secure socket layer - Client-side certificates - Trusted Systems : Memory protection.

Text Book(s)
Stallings William, Cryptography and Network Security: Principles and Practice, 7th Edition, Pearson/Prentice- Hall, 2018.

Reference(s)

Forouzan B A, Cryptography and Network Security, Special Indian Edition, Tata McGraw Hill, 2007.
Padmanabhan TR, Shyamala C K, and Harini N, Cryptography and Security, First Edition, Wiley India Publications, 2011.

Evaluation Pattern:

Assessment	Internal	External
Periodical 1 (P1)	15	
Periodical 2 (P2)	15	
*Continuous Assessment (CA)	20	
End Semester		50

*CA – Can be Quizzes, Assignment, Projects, and Reports.

19SSK311	SOFT SKILLS III	L-T-P-C: 1-0-3-2

Course Outcomes:

CO # 1 - Soft Skills: At the end of the course, the students will have the ability to prepare a suitable resume (including video resume). They would also have acquired the necessary skills, abilities and knowledge to present themselves confidently. They would be sure-footed in introducing themselves and facing interviews.

CO # 2 - Soft Skills: At the end of the course, the students will have the ability to analyse every question asked by the interviewer, compose correct responses and respond in the right manner to justify and convince the interviewer of one's right candidature through displaying etiquette, positive attitude and courteous communication.

CO # 3 - Aptitude: At the end of the course, students will be able to interpret, critically analyze and solve logical reasoning questions. They will have acquired the skills to manage time while applying methods to solve questions on arithmetic, algebra, logical reasoning, and statistics and data analysis and arrive at appropriate conclusions.

CO # 4 – Verbal: At the end of the course, the students will have the ability to understand and use words, idioms and phrases, interpret the meaning of standard expressions and compose sentences using the same.

CO # 5 - Verbal: At the end of the course, the students will have the ability to decide, conclude, identify and choose the right grammatical construction.

CO # 6 – Verbal: At the end of the course, the students will have the ability to examine, interpret and investigate arguments, use inductive and deductive reasoning to support, defend, prove or disprove them. They will also have the ability to create, generate and relate facts / ideas / opinions and share / express the same convincingly to the audience / recipient using their communication skills in English.

Team work: Value of team work in organisations, definition of a team, why team, elements of leadership, disadvantages of a team, stages of team formation. Group development activities: Orientation, internal problem solving, growth and productivity, evaluation and control. Effective team building: Basics of team building, teamwork parameters, roles, empowerment, communication, effective team working, team effectiveness criteria, common characteristics of effective teams, factors affecting team effectiveness, personal characteristics of members, team structure, team process, team outcomes.

Facing an interview: Foundation in core subject, industry orientation / knowledge about the company, professional personality, communication skills, activities before interview, upon entering interview room, during the interview and at the end. Mock interviews.

Advanced grammar: Topics like parallel construction, dangling modifiers, active and passive voices, etc.

Syllogisms, critical reasoning: A course on verbal reasoning. Listening comprehension advanced: An exercise on improving listening skills.

Reading comprehension advanced: A course on how to approach advanced level of reading, comprehension passages. Exercises on competitive exam questions.

Problem solving level IV: Geometry; Trigonometry; Heights and distances; Co-ordinate geometry; Mensuration.

Specific training: Solving campus recruitment papers, national level and state level competitive examination papers;

Speed mathematics; Tackling aptitude problems asked in interview; Techniques to remember (In mathematics). Lateral thinking problems. Quick checking of answers techniques; Techniques on elimination of options, estimating and predicting correct answer; Time management in aptitude tests; Test taking strategies.

TEXTBOOK(S)

A Communicative Grammar of English: Geoffrey Leech and Jan Svartvik. Longman, London.
Adair. J., (1986), "Effective Team Building: How to make a winning team", London, U.K: Pan Books.
Gulati. S., (2006) "Corporate Soft Skills", New Delhi, India: Rupa & Co.
The Hard Truth about Soft Skills, by Amazone Publication.
Data Interpretation by R. S. Aggarwal, S. Chand
Logical Reasoning and Data Interpretation – Niskit K Sinkha
Puzzles – Shakuntala Devi
Puzzles – George J. Summers.

REFERENCE(S)

Books on GRE by publishers like R. S. Aggrawal, Barrons, Kaplan, The Big Book, and Nova.
More Games Teams Play, by Leslie Bendaly, McGraw-Hill Ryerson.
The BBC and British Council online resources
Owl Purdue University online teaching resources

www.the grammarbook.com - online teaching resources www.englishpage.com- online teaching resources and other useful websites.

19CSE335	ETHICAL HACKING	L-T-P-C: 3-0-0-3

Pre-Requisite(s):19MAT115 Discrete Mathematics, 19CSE102 Computer Programming, **19CSE331** Cryptography

Course Objectives

- This course introduces the concepts of Ethical Hacking and gives opportunity to learn about different tools and techniques in Ethical hacking and security.

Course Outcomes

CO1: Understand and apply the core concepts related to malware and software vulnerabilities and their causes
CO2: Appreciate the Cyber Laws and ethics behind hacking and vulnerability disclosure
CO3: Exploit the vulnerabilities related to data and storage systems using state of the art tools and technologies
CO4: Exploit the vulnerabilities related to computer system and networks using state of the art tools and technologies

CO-PO Mapping

PO/ PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1										3	2
CO2	1	2	2	1									3	2
CO3	1	2	2	2	2			2					3	2
CO4	3	3	2	2	2	2		2					3	2

Syllabus

Unit 1
Introduction: Understanding the importance of security, Concept of ethical hacking and essential Terminologies- Threat, Attack, Vulnerabilities, Target of Evaluation, Exploit.

Unit 2
Phases involved in hacking: Foot printing, Scanning, System Hacking, Session Hijacking. Buffer Overflows: Significance of Buffer Overflow Vulnerability, Why Programs/Applications are vulnerable. Reasons for Buffer Overflow Attacks. Methods of ensuring that buffer overflows are trapped. Sniffers: Active and passive sniffing. ARP poisoning and countermeasures. Man in the middle attacks, Spoofing and Sniffing attacks. Sniffing countermeasures. SQL Injection: Attacking SQL Servers, Sniffing, Brute Forcing and finding Application Configuration Files, Input validation attacks. Preventive Measures. Web Application Threats, Web Application Hacking, Cross Site Scripting / XSS Flaws / Countermeasures Correct Web Application Set-up.

Unit 3
Web Application Security: Core Defence Mechanisms. Handling User Access, Authentication, Session Management, Access Control. Web Application Technologies: HTTP Protocol, Requests, Responses and Methods. Encoding schemes. Server side functionality technologies (Java, ASP, PHP). Attacking Authentication: Attacking Session Management, Design Flaws in Authentication Mechanisms Attacking Forgotten Password Functionality, attacking Password change functions. Countermeasures to authentication attacks. Attacking other users: Reflected XSS

Vulnerabilities, Stored XSS Vulnerabilities, DOM-Based XSS Vulnerabilities, HTTP Header Injection. Countermeasures to XSS.

Text Book(s)

Patrick Engebretson. The Basics of Hacking and Penetration Testing, Elsevier; 2013.
Graves K. Ceh: Official certified ethical hacker review guide: Exam 312-50. John Wiley & Sons; 2007..

Reference(s)

Ali S, Heriyanto T. BackTrack 4: Assuring Security by Penetration Testing: Master the Art of Penetration Testing with BackTrack. Packt Publishing Ltd; 2011 Apr 14.
Khare R. Network Security and Ethical Hacking. Luniver Press; 2006.

Evaluation Pattern:

Assessment	Internal	External
Periodical 1 (P1)	15	
Periodical 2 (P2)	15	
*Continuous Assessment (CA)	20	
End Semester		50

*CA – Can be Quizzes, Assignment, Projects, and Reports

19CSE357	BIG DATA ANALYTICS	L-T-P-C: 3-0-0-3

Course Objectives

- The aim of this course is to provide depth knowledge about Big data Technologies and tools used for Big data.
- The students will learn to implement and work on tools to handle large volume of data in parallel and distributed environments. Retrieval and analysis of unstructured data are done using NOSQL databases.

Course Outcomes

CO1: Understand fundamental concepts of Big Data and its technologies
CO2: Apply concepts of MapReduce framework for optimization
CO3: Analyze appropriate NoSQL database techniques for storing and processing large volumes of structured and unstructured data
CO4: Apply data analytics solutions using Hadoop ecosystems
CO5: Explore modern reporting tools for Machine learning

CO-PO Mapping

PO/ PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1							1			3	2
CO2	3	3	2	2	3	2	2	2	2	2			3	2
CO3	3	3	2	2	3	2	2	2	2	2			3	2
CO4	3	3	2	2	3	2	2	2	2	2			3	2
CO5	2			2	3		2	3	2	2			3	2

Syllabus

Unit 1
Introduction to Big Data: Types of Digital Data - Characteristics of Data – Evolution of Big Data - Definition of Big Data - Challenges with Big Data-3Vs of Big Data - Non Definitional traits of Big Data - Business Intelligence vs. Big Data - Data warehouse and Hadoop environment - Coexistence. Big Data Analytics: Classification of analytics - Data Science - Terminologies in Big Data - CAP Theorem - BASE Concept. NoSQL: Types of Databases – Advantages – NewSQL - SQL vs. NOSQL vs NewSQL. Introduction to Hadoop: Features – Advantages – Versions - Overview of Hadoop Eco systems - Hadoop distributions - Hadoop vs. SQL – RDBMS vs. Hadoop - Hadoop Components – Architecture – HDFS - Map Reduce: Mapper – Reducer – Combiner – Partitioner – Searching – Sorting - Compression. Hadoop 2 (YARN): Architecture - Interacting with Hadoop Eco systems.

Unit 2
No SQL databases: Mongo DB: Introduction – Features - Data types - Mongo DB Query language - CRUD operations – Arrays - Functions: Count – Sort – Limit – Skip – Aggregate - Map Reduce. Cursors – Indexes - Mongo Import – Mongo Export. Cassandra: Introduction – Features - Data types – CQLSH - Key spaces - CRUD operations – Collections – Counter – TTL - Alter commands - Import and Export - Querying System tables.

Unit 3

Hadoop Eco systems: Hive – Architecture - data type - File format – HQL – SerDe - User defined functions - Pig: Features – Anatomy - Pig on Hadoop - Pig Philosophy - Pig Latin overview - Data types - Running pig - Execution modes of Pig - HDFS commands - Relational operators - Eval Functions - Complex data type - Piggy Bank - User defined Functions - Parameter substitution - Diagnostic operator. Jasper Report: Introduction - Connecting to Mongo DB - Connecting to Cassandra - Introduction to Machine learning: Linear Regression- Clustering - Collaborative filtering - Association rule mining - Decision tree.

Text Book(s)

Seema Acharya, Subhashini Chellappan, "Big Data and Analytics", Wiley Publication, 2015.

Reference(s)

Hurwitz JS, Nugent A, Halper F, Kaufman M. Big data for dummies. John Wiley & Sons; 2013.
Tom White, "Hadoop: The Definitive Guide", O'Reilly Publications, 2011.
Kyle Banker, "Mongo DB in Action", Manning Publications Company, 2012.
Russell Bradberry, Eric Blow, "Practical Cassandra A developers Approach ", Pearson Education, 2014.

Evaluation Pattern

Assessment	Internal	External
Periodical 1	10	
Periodical 2	10	
*Continuous Assessment (Theory) (CAT)	10	
Continuous Assessment (Lab) (CAL)	40	
End Semester		30

*CA – Can be Quizzes, Assignment, Projects, and Reports.

19CSE445	CLOUD COMPUTING	L-T-P-C: 2-0-3-3

Pre-Requisite(s):19CSE102 Computer Programming, 19CSE301 Computer Networks

Course Objectives

- This course introduces the basic principles of cloud computing, cloud native application development and deployment, containerization principles, micro-services and application scaling.
- It will also equip the students to understand major industry players in the public cloud domain for application development and deployment.

Course Outcomes

CO 1: Understand the basic principles of cloud computing.
CO 2: Apply cloud native application development for containerization and container orchestration.
CO 3: Analyze different types of cloud services – Delivery models, Deployment models.
CO 4: Implement different solution approaches in Cloud – containers in public cloud, setting up private cloud and convert monolithic applications to containers

CO-PO Mapping

PO/PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	1								1		3	2
CO2	3	2	2	2	3	2	3	2	2	2	2	2	3	2
CO3	3	2	2	2	3	2	3	2	1		2		3	2
CO4	3	2	2	2	3	2	3	2	2	2	2	2	3	2

Syllabus

Unit 1
Distributed Computing Taxonomy – Cluster, Grid, P2P, Utility, Cloud, Edge, Fog computing paradigms; Introduction to Cloud Computing – Cloud delivery models (XaaS), Cloud deployment models (Private, Public, Hybrid); Characteristics of Cloud, Major use cases of Cloud; disadvantages and best practices; Major public cloud players in the market; Security Issues and Challenges; Cloud Native application development – Introduction to JavaScript Cloud native application development

Unit 2
Public Cloud – Using public cloud for infrastructure management (compute and storage services), Web application deployment using public cloud services, and Deploying container images in public cloud, Overview of cognitive services, Case study on architecting cloud-based solutions for a chosen scenario

Unit 3
Virtualization – Basics, Cloud vs Virtualization, Types of virtualization, Hypervisor types; Containers – Introduction to dockers and containers, containerization vs virtualization, docker architecture, Use cases, Learn how to build container images, Operations on container images; Kubernetes – Need for orchestration, container orchestration methods, Introduction to Kubernetes, Kubernetes architecture, using YAML file, Running Kubernetes via minikube

Text Book(s)

Rajkumar Buyya et.al. Mastering cloud computing, McGraw Hill Education;2013.

Matthias K, Kane SP. Docker: Up & Running: Shipping Reliable Containers in Production. " O'Reilly Media, Inc."; 2018.

Reference(s)

Kocher PS. Microservices and Containers. Addison-Wesley Professional; 2018.

Sarkar A, Shah A. Learning AWS: Design, build, and deploy responsive applications using AWS Cloud components. Packt Publishing Ltd; 2018.

Menga J. Docker on Amazon Web Services: Build, deploy, and manage your container applications at scale. Packt Publishing Ltd; 2018.

Bentley W. OpenStack Administration with Ansible 2. Packt Publishing Ltd; 2016.

Evaluation Pattern

Assessment	Internal	External
Periodical 1	10	
Periodical 2	10	
*Continuous Assessment (Theory) (CAT)	10	
Continuous Assessment (Lab) (CAL)	40	
End Semester		30

*CA – Can be Quizzes, Assignment, Projects, and Reports.