Mahavir Education Trust's

Shah & Anchor Kutchhi Engineering College

An Autonomous Institute Affiliated to University of Mumbai

2025-26 *Scheme & Syllabus*

Bachelor of Technology (B.Tech.) in Electronics & Computer Science

Third Year : Semester V & VI (With Effect for 2025-2026)

Mahavir Education Trust's

Shah & Anchor Kutchhi Engineering College

An Autonomous Institute Affiliated to University of Mumbai

CURRICULUM STRUCTURE THIRD YEAR B.Tech.

Electronics & Computer Science

REVISION: R1-V0-2025-26

Effective for Academic Year 2025-26

Approved by Board of Studies on 18/06/2025

Approved by Academic Council on 19/06/2025

Dr. Vidyullata Devmane
I/C HOD

Chairman BoS
Department of Electronics & Computer Science
Shah & Anchor Kutchhi Engineering College
Chembur, Mumbai - 400088

Shah & Anchor Kutchhi Engg. College
Autonomous Institute
Mumbai-88

Dr. Bhavesh Patel
Principal

Dr. Bhavesh Patel
Chairman
Academic Council
Shah & Anchor Kutchhi Engineering College
Chembur, Mumbai - 400088

UG Program in Electronics & Computer Science

VISION:

Impart quality education in Electronics and Computer Science Engineering to create world class technocrats and entrepreneurs to meet industry standards.

MISSION:

1. To deliver quality academic program in electronics and computer science engineering.
2. To develop skilled professionals capable of providing Electronics and Computer-based solutions giving emphasis on industrial challenges.
3. To improve employability, entrepreneurship and higher education of electronics and computer science engineers with ethical and professional approach.

PROGRAM SPECIFIC OUTCOMES (PSO)

- **PSO1:** Students will be able to develop financially viable as well as ethical solutions for real world problems by applying the principles of electronics engineering and techniques of computer science.
- **PSO2:** Students will be able to excel in next-generation electronics and computer science domains, driving technological advancements for societal benefit in the area of IoT, Communication Networks, Cloud computing, Artificial Intelligence, Database and Information management.

PROGRAM EDUCATIONAL OBJECTIVES (PEO)

- **PEO1:** To develop a strong foundation of engineering knowledge to build successful engineers.
- **PEO2:** To encourage graduates for higher studies and research activities to accomplish lifetime learning.
- **PEO3:** To inculcate team spirit and leadership qualities in graduates with the ability to become better Entrepreneurs.

UG Program in Electronics and Computer Science

PROGRAM OUTCOMES (PO)

1. **PO1: Engineering Knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

2. **PO2:** Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

3. **PO3:** Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

4. **PO4:** Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis interpretation of data to provide valid conclusions. (WK8).

5. **PO5:** Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

6. **PO6:** The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

7. **PO7:** Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national international laws. (WK9)

8. **PO8:** Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

9. **PO9:** Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.

10. **PO10:** Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

11. **PO11:** Life-Long Learning: Recognize the need for, and have the preparation and abilityfor i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

UG Program in Electronics & Computer Science

KNOWLEDGE AND ATTITUDE PROFILE (WK)

1. **WK1:** A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.
2. **WK2:** Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.
3. **WK3:** A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.
4. **WK4:** Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.
5. **WK5:** Knowledge, including efficient resource use, environmental impacts, whole-life cost,reuse of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.
6. **WK6:** Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.
7. **WK7:** Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.
8. **WK8:** Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.
9. **WK9:** Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding andrespect, and of inclusive attitudes.

UG Program in Electronics & Computer Science

PREAMBLE

To meet the challenge of technological changes and to ensure excellence in engineering education, the Electronics and Computer Science branch stands as pillar of effective NEP implementation, innovation and research. Electronics and Computer Science has been driving progress across IT and Electronics industries and shaping the way we interact with the changing world. ECS journey is embarked with practical applications, theoretical concepts, integrating interdisciplinary skills to nurture future engineers.
At the heart of the electronics domain, ECS branch lays a quest to harness the power of semiconductors and circuits to enhance communication. It incorporates the Internet of things and embedded systems.

In parallel, computer science offers implementation of algorithms, data structures and artificial intelligence. The booming fields of artificial intelligence, deep learning and robotics have been well integrated in the curriculum. All latest technological developments such as industry 4.0, augmented and virtual reality, robotics, drone technology are parts of the curriculum.

In the electronics and computer science curriculum, knowledge of various domains are integrated. Journey of innovations, project based learning, exposure to industries through internships have been initiated. It will lead a path towards a bright future where technology and holistic education will serve as a catalyst for positive change.

Outcome and skill-based education with knowledge of Indian culture and student centric learning are implemented in the revised curriculum. Curriculum has 173 credits in the undergraduate program. Faculty resolved that course objectives and course outcomes are to be clearly defined for each course, so that all faculty members in the institute understand the depth and approach of the course to be taught, which will enhance learner's learning quotient.

Choice based Credit and grading system enables a much-required shift in focus from teacher- centered to learner-centric education. The workload estimated is based on the investment of time in learning and not in teaching. It also focuses on continuous evaluation which will enhance the quality of education.

Mahavir Education Trust's

Shah & Anchor Kutchhi Engineering College

An Autonomous Institute Affiliated to University of Mumbai

Program Structure for Third Year B.Tech. in Electronics & Computer Science (with effect for 2025–2026)

Semester V

Sr. No.	Course Title	Course Code	Learning Scheme					Credits	Assessment Scheme												
			Actual Contact Hrs/Week			SL	Total		Exam	Theory					Practical				SLA		Total
										CIA		ESE	Total		CIAP	ESEP	Total				
			CL	TL	LL		Notional Hours		Duration in Hrs	MSE Max	CCE Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Communication Engineering	ECCR1501	3	1	2	2	120	4	2	20	20	60	100	40	50	-	50	20	25	10	175
2	Computer Organization and Architecture	ECCR1502	3	1	2	2	120	4	2	20	20	60	100	40	50	-	50	20	25	10	175
3	Software Engineering	ECCR1503	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175
4	Web Technologies	ECCR1504	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175
5	Department Level Optional Course - I	ECDLOCR1505x	2	-	4	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125
6	Skill-based Lab IV- Linux Server Administration Lab	ECLR1506	-	-	4	-	60	2	2	-	-	-	-	-	25	25	50	20	-	-	50
Total			**14**	**4**	**16**	**10**	**660**	**22**		**80**	**130**	**240**	**450**		**200**	**100**	**300**		**125**		**875**

Abbreviations:
CL: Classroom Learning TL: Tutorial Learning LL: Laboratory Learning SL: Self Learning (Activity/Assignment/Micro-Project) CIA: Continuous Internal Assessment
CCE: Continuous Comprehensive Evaluation MSE: Mid Semester Examination ESE: End Semester Examination CIAP: Continuous Internal Assessment Practical ESEP: End Semester Examination Practical
SLA: Self Learning Assessment

Note:
1. Notional Learning hours for the semester are (CL+LL+TL+SL) hrs × 15 Weeks. 2. 1 credit is equivalent to 30 Notional hrs.
3. Self learning hours shall not be reflected in the Time Table. 4. SLA will have separate head of passing

Sr.No	Department Level Optional Course - I (ECDLOCR1505x)	Course Code
1	Software Testing and Quality Assurance	ECDLOCR15051
2	System Verification with System Verilog	ECDLOCR15052
3	Signals and Systems	ECDLOCR15053
4	Sensor technologies and applications	ECDLOCR15054
5	Control System and Instrumentation	ECDLOCR15055

Mahavir Education Trust's

Shah & Anchor Kutchhi Engineering College

An Autonomous Institute Affiliated to University of Mumbai

Program Structure for Third Year B.Tech. in Electronics & Computer Science
(with effect from 2025–2026)

Semester VI

Sr. No.	Course Title	Course Code	Learning Scheme: Actual Contact Hrs/Week CL	TL	LL	SL	Total Notional Hours	Credits	Assessment Scheme: Exam Duration in Hrs	Theory: CIA MSE Max	CCE Max	ESE Max	Total Max	Total Min	Practical: CIAP Max	ESEP Max	Total Max	Total Min	SLA Max	SLA Min	Total
1	Embedded System Design	ECCR1601	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175
2	Artificial Intelligence: Knowledge Representation and Reasoning	ECCR1602	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175
3	Computer Networks	ECCR1603	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175
4	Data Warehousing and Mining	ECCR1604	3	1	2	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125
5	Department Level Optional Course -II	ECDLOCR1605x	2	-	4	2	120	4	2	20	20	60	100	40	50	-	50	20	25	10	175
6	Professional Communication and Ethics-II	ECLR1606	-	-	4	-	60	2	-	-	-	-	-	-	50	-	50	20	-	-	50
Total			**14**	**4**	**16**	**10**	**660**	**22**		**80**	**130**	**240**	**450**		**200**	**100**	**300**		**125**		**875**

Abbreviations:
CL: Classroom Learning TL: Tutorial Learning LL: Laboratory Learning SL: Self Learning (Activity/Assignment/Micro-Project) CIA: Continuous Internal Assessment
CCE: Continuous Comprehensive Evaluation MSE: Mid Semester Examination ESE: End Semester Examination CIAP: Continuous Internal Assessment Practical ESEP: End Semester Examination Practical
SLA: Self Learning Assessment

Note:
1. Notional Learning hours for the semester are (CL+LL+TL+SL) hrs × 15 Weeks. 2. 1 credit is equivalent to 30 Notional hrs.
3. Self learning hours shall not be reflected in the Time Table. 4. SLA will have separate head of passing

Sr.No	Department Level Optional Course – II (ECDLOCR1605x)	Course Code
1	Machine Learning	ECDLOCR16051
2	Industrial Automation	ECDLOCR16052
3	Information Theory and Coding	ECDLOCR16053
4	Idea to Product Expedition	ECDLOCR16054
5	Fundamentals of Electric Vehicle	ECDLOCR16055
6	Digital Signal Processing	ECDLOCR16056

Semester V

Program	Third Year B.Tech Electronics & Computer Science(Semester V)	CL	TL	LL	SL	C
Course Name: Communication Engineering	**Course Code:** ECCR1501	3	1	2	2	4
Course Type:	-					
Pre-requisite:	Basic courses in 1.Engineering Mathematics 2.Electronic Devices and Applications					

I RATIONALE

Communication Engineering is a critical and evolving field that underpins the infrastructure of modern society. With the rapid advancement of technology, the demand for efficient, secure, and high-speed communication systems has never been greater. Studying this course offers both theoretical foundations and practical skills necessary for designing, analyzing, and improving various communication systems. Communication engineering is essential to mobile networks, the internet, satellite communication, wireless technologies, and IoT systems. Mastery of these systems allows engineers to contribute to cutting-edge innovations like 5G, 6G, and beyond.

II COMPETENCY

Demonstrate the applications of the fundamental principles of analog and digital communication systems, including modulation techniques, noise analysis, and system components, to analyze and design basic communication networks

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Apply the principles of analog modulation in understanding communication systems
2	Apply the concepts of pulse modulation and multiplexing in communication engineering
3	Analyze different pulse shaping procedures and the generation of Inter symbol Interference
4	Evaluate different digital modulation systems

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total	Credits		Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA		ESE	Total		CIAP	ESEP	Total				
								Notional		Duration	MSE	CCE										
				CL	TL	LL		Hours		in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Communication Engineering	-	ECCR1501	3	1	2	2	120	4	2	20	20	60	100	40	50	-	50	20	25	10	175

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 10)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Apply the principles of amplitude modulation and demodulation to study AM circuits	Amplitude Modulation and demodulation: (AM)	CO1
2	Apply the concepts of frequency modulation and demodulation to study FM circuits	Frequency Modulation and demodulation: (FM)	CO1
3	Apply the techniques of pulse amplitude, width, and position modulation to generate and analyze modulated signals	Pulse modulation (PAM/PWM/PPM).	CO2
4	Apply the principles of time division multiplexing to implement and observe the transmission of multiple signals over a shared channel	Time division multiplexing	CO2
5	Apply the concept of frequency division multiplexing to combine multiple signals within distinct frequency bands	Frequency division multiplexing	CO2
6	Apply the sampling theorem to demonstrate the conditions required for accurate signal reconstruction	Sampling theorem	CO2
7	Apply various line coding techniques such as NRZ, RZ, and Manchester to encode binary data	Generation of Line codes	CO3
8	Apply the principles of BASK to generate and detect amplitude-shift keyed signals	Binary Amplitude Shift Keying (BASK)	CO4
9	Apply the techniques of BPSK signal generation and coherent detection using appropriate hardware or simulation tools	Binary Phase Shift Keying (BPSK)	CO4
10	Apply the principles of BFSK to implement and analyze frequency-shift keyed signal transmission and detection	Binary Frequency Shift Keying(BFSK)	CO4
11	Apply simulation tools to model amplitude and frequency modulation systems	Simulation of AM and FM	CO1
12	Apply simulation techniques to generate and analyze pulse amplitude, position, and width modulated signals	Simulation of PAM, PPM, PWM	CO2
13	Apply digital modulation techniques using simulation tools to generate and analyze BPSK, BASK, and MSK signals	Simulation of BPSK/BASK/MSK modulation	CO4
14	Apply the principles of correlative coding to simulate duobinary encoding and decoding processes	Simulation of duobinary encoder, decoder	CO3

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Understand the EM spectrum and analog communication system block diagram; apply modulation concepts 2. Understand different types of noise and apply concepts like signal-to-noise ratio, noise factor, noise figure, and noise temperature to assess system performance 3. Understand and apply the mathematical and graphical representation of AM waves, analyze their frequency spectrum, power distribution, and behavior with complex modulating signals 4. Understand and apply the principles of angle modulation, including FM and PM theory, spectrum analysis of FM, and comparison with AM	**Analog Modulation Systems** 1.1 Electromagnetic Spectrum Block diagram of Analog communication system. Need for modulation 1.2 Types of Noise, Signal-to-noise ratio, Noise factor, Noise Figure, Noise Temperature 1.3 Representation of AM wave (Mathematical and Graphical), Frequency spectrum of AM wave, AM Power Distribution, AM for a Complex Modulating Signal 1.4 Principles of Angle Modulation: Theory of Frequency Modulation (FM) and Phase Modula.tion (PM) - Basic Concepts, Spectrum Analysis of FM Wave, Comparison of AM, FM and PM	14
2	CO2	1. Understand and apply the sampling theorem and signal quantization for accurate analog-to-digital conversion 2. Understand and apply the principles of generation and detection of PAM, PCM, and Delta Modulation in digital communication systems 3. Understand and apply multiplexing techniques including TDM with T1 carrier system and FDM for efficient signal transmission	**Pulse Modulation** 2.1 Sampling theorem and quantization of signals 2.2 Generation and Detection of Pulse Amplitude Modulation (PAM). Pulse Code Modulation (PCM), and Delta Modulation (DM) 2.3 Multiplexing Techniques: Time Division Multiplexing (TDM): T1 carrier system, Frequency Division Multiplexing (FDM).	8

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Understand and apply line coding techniques, their desirable properties, and analyze the power spectral density (PSD) of digital signals 2. Apply the concepts of ISI, eye diagram, quality factor, BER, and Nyquist bandwidth; understand their impact on digital communication performance 3. Apply types of equalizers including linear equalizer, and correlative coding techniques like duo-binary and modified duo-binary; understand their role in mitigating ISI	**Pulse Shaping** 3.1 Line codes and their desirable properties, PSD of digital data 3.2 Concept of Inter symbol interference (ISI), Eye diagram: Quality Factor and BER, Nyquist Bandwidth 3.3 Types of equalizers: Linear equalizer. Correlative coding: Duo-binary encoding and modified duo-binary encoding	9
4	CO4	1. Evaluate and apply the concepts of generation, detection, signal space diagram, power spectral density, error probability, and spectral efficiency of BPSK 2. Evaluate and apply the principles of QPSK and M-ary PSK, including their generation, detection, error performance, and spectral efficiency 3. Evaluate and apply the generation, detection, error performance, and spectral characteristics of Binary Amplitude Shift Keying (BASK) 4. Evaluate and apply the concepts of generation, detection, and performance analysis of Quadrature Amplitude Modulation (QAM) 5. Evaluate and apply the generation, detection, error analysis, and spectral properties of Binary Frequency Shift Keying (BFSK) 6. Evaluate and apply the principles of Minimum Shift Keying (MSK), including its generation, detection, and spectral efficiency	**Digital Modulation** 4.1 Generation, detection, signal space diagram, power spectral density, Error probability and spectrum efficiency analysis of Binary Phase Shift Keying (BPSK) 4.2 Quaternary Phase Shift Keying (QPSK), M-ary PSK 4.3 Binary Amplitude Shift Keying (BASK), 4.4 Quadrature Amplitude Modulation (QAM) 4.5 Binary Frequency Shift Keying (BFSK) 4.6 Minimum Shift Keying (MSK).	14

VII SUGGESTED SELF LEARNING ACTIVITIES

1.Mini Project
2.Case Study
3.Technical Paper study and presentation
4.Assignments and presentations on advanced topics

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Apply the principles of analog modulation in understanding communication systems	-	15%	40%	25%	10%	10%	100%
2	Apply the concepts of pulse modulation and multiplexing in communication engineering	-	15%	40%	25%	10%	10%	100%
3	Analyze different pulse shaping procedures and the generation of Inter symbol Interference	-	10%	30%	35%	15%	10%	100%
4	Evaluate different digital modulation systems	-	10%	30%	35%	15%	10%	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Communication System	4th Edition	Simon Haykin	John Wiley and Sons	2000
2	Principles of Communication Systems	3rd Edition	Taub and Schilling	Tata Mc-Graw Hill	2007
3	Electronics Communication System	4th Edition	Kennedy and Davis	Tata McGraw Hill	1999
4	Analog and Digital Communication	1st Edition	T. L. Singal	Tata Mc-Graw Hill	2012
5	Digital Communication: Fundamentals and Applications	2nd Edition	Sklar B, and Ray P. K	Pearson India	2009

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Digital communication	4th Edition	Simon Haykin	John Wiley and sons	2014
2	Electronics Communication Systems	3rd Edition	Wayne Tomasi	Pearson Education	2001
3	Analog and Digital Communication	2nd Edition	R P Singh and S. Sapre	Tata McGraw Hill	2007
4	Communication System Engineering	2nd Edition	Proakis and Salehi	Pearson Education	2001

Online References

Sr. No	Website Name
A	https://tinyurl.com/mit602, MIT Open courseware;
B	https://tinyurl.com/mit6450, MIT Open courseware;
C	https://tinyurl.com/mit1636, MIT Open courseware
D	https://nptel.ac.in/courses/117102059, NPTEL course
E	https://archive.nptel.ac.in/courses/117/105/117105144/, NPTEL course

Program	Third Year B.Tech Electronics & Computer Science (Semester V)	CL	TL	LL	SL	C
Course Name: Computer Organization and Architecture	**Course Code:ECCR1502**	3	1	2	2	4
Course Type:	-					
Pre-requisite:	Digital Electronics					

I RATIONALE

Computer Organization and Architecture is a core subject in computer science and engineering that provides foundational knowledge about how computer systems work at the hardware level. Understanding this subject is essential for designing efficient systems, optimizing software, and solving real-world computational problems. It connects how high-level programming languages and operating systems interact with underlying hardware.

II COMPETENCY

Analysis of execution cycles, architecture types, performance evaluation, execution cycles, architecture types, performance evaluation , Comparing architecture efficiency, hazard detection in pipelining, performance analysis.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Examine the design and architecture of computer systems.
2	Analyze memory hierarchy and parallelism.
3	Analyze the objectives and functions of an Operating System
4	Discuss the various architectural enhancements in modern processors

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total	Credits		Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code					Notional		Exam	CIA		ESE	Total		CIAP	ESEP	Total				
				CL	TL	LL		Hours		Duration in Hrs	MSE Max	CCE Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Computer Organization and Architecture	-	ECCR1502	3	1	2	2	120	4	2	20	20	60	100	40	50	-	50	20	25	10	175

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Simulate and analyze how direct-mapped cache memory works using ParaCache.	Direct mapping Cache using Para Cache simulator.	CO2
2	Compare direct vs associative mapping using simulations and evaluate hit/miss performance.	Associative Mapped Cache using Para Cache simulator.	CO2
3	Implement and test the LRU page replacement algorithm.	Program for LRU page replacement algorithm.	CO2
4	Apply FIFO algorithm to simulate virtual memory page replacement.	Program for FIFO page replacement algorithm.	CO2
5	Simulate and compare FCFS and SJF CPU scheduling in terms of waiting and turnaround times.	Implementation of CPU Scheduling. 1)FCFS 2) SJF	CO3
6	Implement and evaluate advanced CPU scheduling techniques including RTF and Priority scheduling	Implementation of CPU Scheduling-1) Remaining Time First and 2) Priority based 1) Remaining Time First and 2) Priority based	CO3
7	Design and simulate disk scheduling algorithms and measure seek times.	Programs to simulate implementation of Disk Scheduling Algorithms: FCFS, SSTF.	CO3
8	Apply Banker's Algorithm to detect and prevent deadlocks in resource allocation.	Program to simulate Bankers Algorithm for Deadlock Detection.	CO4
9	Simulate file organization structures and differentiate their performance and accessibility.	Program to simulate the following file organization techniques a) Single level directory b) Two level directory	CO4
10	Identify and classify different types of pipeline hazards in instruction sets.	Program to detect the type of Hazards for a set of instructions	CO2

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Define and explain performance metrics such as Clock Speed, CPI, MIPs, MFLOPs. 2. Illustrate integer and floating-point number representations. 3. Analyze the functioning of CPU architecture and instruction cycles. 4. Differentiate between hardwired and microprogrammed control units. 5. Compare RISC and CISC architectures in terms of complexity and efficiency. 6. Design basic control units using micro-instruction formats.	**Introduction to Computer Organization and Architecture** 1.1 Fundamental Units of a Computer, Basic Measures of Computer Performance –Clock Speed, CPI, MIPs and MFlops 1.2 Number Representation methods- Integer and Floating-point 1.3 CPU Architecture, Register Organization, Instruction cycle, Instruction Formats, Control Unit Design- Hardwired and Micro-programmed Control: Vertical and Horizontal Micro-Instructions 1.4 Nano-programming, Comparison between CISC and RISC architectures	8

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
2	CO2	1. Classify different types of memory and explain memory hierarchy. 2. Compare cache mapping techniques and write policies. 3. Explain the concept of virtual memory and its management using paging/segmentation. 4. Apply page replacement algorithms to optimize memory utilization. 5. Identify types of I/O devices and their access methods. 6. Distinguish between types of buses and evaluate arbitration methods. 7. Describe Flynn's classification and interpret Amdahl's law for speedup. 8. Analyze pipeline performance and identify various types of hazards.	**Parallelism and Memory Hierarchy** 2.1 Classification of Memories-Primary and Secondary Memories, ROM and RAM, Memory Inter- leaving Memory Hierarchy, Cache Memory Concepts, Mapping Techniques, Write Policies, Cache Coherency. 2.2 Virtual Memory Management-Concept, Segmentation, Paging, Page Replacement policies. 2.3 Types of I/O devices and Access methods, Types of Buses, Bus Arbitration. 2.4 Introduction to Parallel Processing Concepts, Flynn's classification, Amdahl's law. 2.5 Pipelining - Concept, Speedup, Efficiency, Throughput, Types of Pipeline hazards and solutions.	15

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Define and describe the concept of a process and process states. 2. Illustrate the structure and role of the Process Control Block. 3. Compare and evaluate process scheduling algorithms. 4. Explain the concept and benefits of multithreading in systems. 5. Apply IPC mechanisms to coordinate process execution. 6. Analyze process synchronization issues and propose solutions. 7. Identify and explain deadlock conditions and recommend prevention methods.	**Operating System Concepts** 3.1 Concept of a Process, Process States, Process Description, Process Control Block 3.2 Process scheduling -Pre-emptive and Non preemptive scheduling algorithms (FCFS, Priority, SJF), Concept of Multithreading 3.3 Inter-Process Communication, Process Synchronization, Deadlock and Prevention 3.4 File Management -File Organization and Access 3.5 I/O Management and Disk Scheduling: FCFS, SSTF	17

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
4	CO4	1. Describe the concept and operation of superscalar architectures. 2. Compare in-order vs out-of-order execution with examples. 3. Analyze the performance implications of using clusters vs multi-core CPUs. 4. Differentiate between GPU and CPU architectures based on use-case. 5. Evaluate the suitability of GPUs for different types of workloads. 6. Apply concepts of parallel computing using clusters or GPUs in problem-solving.	**Architectural Enhancements** 4.1 4.1 Superscalar Architectures, Out-of-Order Execution, Multi-core processors, Clusters, GPU	5

VII SUGGESTED SELF LEARNING ACTIVITIES

Mini Project / Case study

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Examine the design and architecture of computer systems.	-	-	60%	40%	-	-	100%
2	Analyze memory hierarchy and parallelism.	-	-	50%	50%	-	-	100%
3	Analyze the objectives and functions of an Operating System.	-	-	50%	0%	-	-	100%
4	Discuss the various architectural enhancements in modern processors.	-	-	40%	60%	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Computer Organization and Architecture: Designing for Performance	8th Edition	William Stallings	Pearson Publication	2010
2	Computer Organization and Embedded Systems	6th Edition	C. Hamacher, Z. Vranesic and S. Zaky	McGraw Hill	2010
3	Operating System: Internals and Design Principles	8th Edition	William Stallings	Prentice Hall	2014
4	Operating System Concepts	9th Edition	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne	John WileySons	2015

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Computer Architecture and Organization	3rd Edition	P.Hayes	McGraw-Hill	2019
2	Computer Organization and Design – The Hardware/Software Interface	4th Edition	D. A. Patterson and J. L. Hennessy	Morgan Kaufmann	2014
3	Operating Systems	3rd Edition	Achyut Godbole and Atul Kahate	McGraw Hill Education	2011
4	Operating System Design and Implementation	4th Edition	Andrew Tannenbaum	Pearson	2015

Online References

Sr. No	Website Name
A	https://archive.nptel.ac.in/courses/106/105/106105163/, NPTEL course Computer Organization and Architecture

Program	Third Year B.Tech. Electronics & Computer Science (SemesterV)	CL	TL	LL	SL	C
Course Name: Software Engineering	**Course Code:** ECCR1503	3	1	2	2	4
Course Type:	-					
Pre-requisite:	C Programming, Java or C++ programming					

I RATIONALE

The Software Engineering course is designed to equip students with the foundational knowledge and practical skills necessary to develop reliable, scalable, and maintainable software systems. With software playing a pivotal role across industries, this course introduces students to essential software process models, including both traditional approaches like the Waterfall model and modern agile methodologies such as Scrum and Extreme Programming. It emphasizes the importance of requirement engineering and design thinking, enabling students to accurately capture system needs and translate them into robust architectural and component-level designs using tools like UML. Through detailed modules on project scheduling, cost estimation, and risk management, students gain proficiency in planning, tracking, and executing software projects effectively. The course also covers software quality assurance, risk handling, and reliability practices aligned with industry standards. Finally, it provides hands-on exposure to software testing techniques, including white-box and black-box testing, to ensure the delivery of high-quality software products. Overall, the subject bridges theoretical foundations with industry-relevant practices, preparing students for diverse roles such as software developers, project managers, system analysts, and QA engineers in the rapidly evolving software industry.

II COMPETENCY

Apply and select suitable software process models based on project needs.
Apply requirement engineering and design principles to build effective system models and architecture.
Plan and manage software projects through scheduling, estimation, and risk assessment techniques.
Ensure software quality through structured testing methods and validation approaches

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Apply software engineering concepts to choose process models for software project development.
2	Write software requirement specification (SRS) for software system which serves as the foundation for the system.
3	Generate the project schedule for estimation and risks mitigation for the software system.
4	Apply testing strategies and tactics for software system.

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total	Credits		Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA		ESE	Total		CIAP	ESEP	Total				
								Notional		Duration	MSE	CCE										
				CL	TL	LL		Hours		in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Software Engineering	-	ECCR1503	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Students will be able to define and formulate clear and concise problem statements based on real-world needs	To prepare problem statement for any project.	CO1
2	Students will be able to create a structured SRS document by analyzing user and system requirements	Preparation of Software Requirement Specification (SRS) document	CO2
3	Students will be able to develop UML use case diagrams and illustrate scenarios to represent system functionality from the user's perspective.	Modeling UML use case diagrams and capturing use case scenarios	CO2
4	Students will be able to analyze the system domain and construct UML class diagrams to show structure and relationships.	Identifying domain classes and modeling UML class diagrams	CO2
5	Students will be able to design state chart diagrams to represent object behavior across states and events.	Study and draw behavioural modeling using state chart diagrams	CO2
6	Students will be able to illustrate sequence diagrams to model interaction between system components in time sequence	Study and draw behavioural modeling using sequence diagrams	CO2
7	Students will be able to construct Level 0 and Level 1 DFDs to represent system data processing and flow.	Draw the data flow diagrams at level 0 and level 1	CO2
8	Students will be able to develop activity diagrams that model workflows for system use cases	Draw activity diagram of all use cases.	CO2
9	Students will be able to model collaboration diagrams to visualize interactions and relationships among objects.	Draw collaboration diagram of all use cases	CO2
10	Students will be able to create project scheduling charts using tools like Project Libre for effective project tracking.	Creation of Time-Line charts using Project Libre tool	CO3
11	Students will be able to apply testing techniques to validate and verify software systems using structured strategies.	Testing strategies and tactics for software system	CO4

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Explain the nature of software and compare various prescriptive and evolutionary software process models, selecting appropriate models for specific project contexts. 2. Analyze and differentiate between specialized software process models, particularly Agile methodologies like XP and Scrum, and evaluate their applicability in dynamic project environments.	**Introduction to Software Engineering, Process Models and Requirement Engineering** 1.1 Nature of Software, Software Process framework, Prescriptive Models: Waterfall Model, Incremental, RAD Models Evolutionary Process Models: Prototyping, Spiral and Concurrent Development Model. 1.2 Specialized Models: Component based. Agile process, Agility Principles, Extreme Programming (XP), Scrum.	10
2	CO2	1. Classify different types of software requirements, prepare a Software Requirement Specification (SRS) document, and develop use case models using UML. 2. Apply design principles and concepts to develop architectural, component-level, and system-level software designs, including user interface design.	**Modelling and Design Engineering** 2.1 Types of Requirements, Requirement Specification (SRS), Developing Use Cases (UML) Engineering Task, 2.2 Design Concepts, Design Principles. Architecture Design, Component Level Design, System Level Design, User Interface Design.	13

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Create project schedules using Gantt and PERT charts and define appropriate task sets for managing software development activities. 2. Estimate project size, effort, and cost using decomposition techniques and empirical models like LOC, Function Point, Use Case–based estimation, and COCOMO II. 3. Identify various types of software project risks and perform risk assessment, projection, and develop Risk Mitigation, Monitoring, and Management (RMMM) plans. 4. Evaluate software quality using McCall's Quality Factors, conduct Formal Technical Reviews (FTRs), and apply software reliability and walkthrough techniques.	**Project scheduling, Cost Estimation, Software Risk & Quality Management** 3.1 Project Scheduling, defining a Task Set for the Software Project, Gantt charts, Program Evaluation Review Techniques (PERT), Tracking the Schedule. Software Project Estimation, Decomposition Techniques, LOC based, FP based and Use case-based estimations, Empirical estimation Models. COCOMO II Model. 3.2 Software Risk, Types of Risk, Risk Identification, Risk Assessment, Risk Projection, RMMM Software Quality Assurance Task and Plan, McCall's Quality Factors, Software Reliability, Formal Technical Review (FTR), Walkthrough	13

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
4	CO4	1. Explain software testing fundamentals and apply appropriate testing strategies for conventional and object-oriented software architectures. 2. Design and perform unit testing and integration testing to verify the correctness of software modules. 3. Differentiate between validation testing and system testing, and select suitable methods for end-to-end verification of software systems. 4. Apply various testing tactics including White-Box (Basis Path, Control Structure Testing) and Black-Box testing to detect and resolve software defects.	**Software Testing Strategies and Tactics** 4.1 Software Testing Fundamentals, Testing strategies for conventional and Object Oriented architectures, Unit testing, Integration 4.2 Validation and System Testing. Testing Tactics: White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing.	9

VII SUGGESTED SELF LEARNING ACTIVITIES

Case study/ Research Paper survey / Microproject/Literature Review Report and Presentation

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Apply software engineering concepts to choose process models for software project development.	-	-	80%	20%	-	-	100%
2	Write software requirement specification (SRS) for software system which serves as the foundation for the system.	-	-	60%	40%	-	-	100%
3	Generate the project schedule for estimation and risks mitigation for the software system.	-	-	60%	40%	-	-	100%
4	Apply testing strategies and tactics for software system.	-	-	60%	40%	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Software Engineering: A Practitioner's Approach	8th Edition	Roger S Pressman	McGrawHill	2010
2	An integrated approach to Software Engineering	3rd Edition	Pankaj Jalote	(Narosa Edition) Springer Edition	2005

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Software Engineering	10th Edition	Ian Sommerville	Pearson Education	2015
2	Fundamentals of Software Engineering	5th Edition	Rajib Mall	Prentice Hall India	2018
3	Software Engineering	1st Edition	Jibitesh Mishra and Ashok Mohanty	Pearson edition	2016
4	Software Engineering Principles and Practice	3rd Edition	Hans Van Vilet	Wiley	2008

Online References

Sr. No	Website Name
A	https://onlinecourses.nptel.ac.in/noc24_cs119/preview, NPTEL Course: Software Engineering By Prof. Rajib Mall (IIT Kharagpur);
B	https://www.mlsu.ac.in/econtents/16EBOOK-7th_ed_software_engineering_a_practitioners_approach_by_roger_s._pressman_.pdf, Text Book;
C	https://davcollegetitilagarh.org/wp-content/uploads/2020/09/fundamentals-of-software-engineering-fourth-edition-rajib-mall.pdf, Book;

Program	Third Year B.Tech Electronics & Computer Science (Semester V)	CL	TL	LL	SL	C
Course Name: Web Technologies	**Course Code:** ECCR1504	3	1	2	2	4
Course Type:	-					
Pre-requisite:	knowledge of programming languages like C, Java					

I RATIONALE

Learning web technologies is beneficial due to its high demand in the job market, diverse career paths, and potential for both personal and professional growth. It equips individuals with skills applicable to various industries and provides a foundation for innovation and entrepreneurship.

II COMPETENCY

Design responsive web interfaces using HTML, CSS, and JavaScript; develop dynamic and interactive web applications using server-side technologies such as PHP or frameworks like Laravel; and implement database connectivity, user authentication, and MVC architecture.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Design static web pages using HTML5.
2	Design layout of web pages using CSS3 and Bootstrap.
3	Apply the concepts of client-side validation and scripts to static web pages using JavaScript and JQuery and build responsive web pages using front-end framework Bootstrap
4	Develop a web application using appropriate web development framework

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total	Credits	Theory						Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA		ESE	Total		CIAP	ESEP	Total				
								Notional		Duration	MSE	CCE										
				CL	TL	LL		Hours		in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Web Technologies	-	ECCR1504	3	1	2	2	120	4	2+2	20	20	60	100	40	25	25	50	20	25	10	175

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Design and develop static web pages using HTML and CSS to create a basic online bookstore interface.	Design static web pages required for an online book store web site.	CO1, CO2
2	Implement JavaScript-based client-side validation for web forms .	Write JavaScript to validate the fields of the Registration page.	CO1, CO2, CO3
3	Demonstrate the use of CSS to style HTML web pages	Create a webpage using inline, internal and external CSS styling.	CO2
4	Implement JavaScript programs using functions and pop-up boxes	Develop JavaScript with POP-UP boxes and functions	CO1, CO2, CO3
5	Design an interactive HTML page	Write an HTML page that contains a selection box with a list of 5 countries.	CO1, CO2
6	Develop an interactive HTML page integrated with JavaScript	Write an HTML page that takes a number from text field in the range of 0 to 999 and shows it in words.	CO1, CO3
7	Design and implement PHP scripts to perform server-side processing tasks	Write a PHP Script to find out the Sum of the Individual Digits.	CO3
8	Create and parse an XML document using Java to retrieve user details	Create an XML document that contains 10 users' information.	CO3
9	implement dynamic web applications using PHP, Servlets, and JSP technologies	Implement the web applications using PHP, Servlets, JSP.	CO4
10	Develop a web application using JavaScript to create, store, and retrieve cookies,	Develop A web application that lists all cookies stored in the browser on clicking "List Cookies" button. .	CO4

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Design web pages using HTML5 frames and hyperlinks, enabling navigation between internal and external resources 2. Insert and manage multimedia elements to enhance user experience on web pages 3. Develop interactive HTML5 forms using form elements	**Introduction to HTML5** 1.1 Basic structure of an HTML5 document and Creating an HTML5 document 1.2 Mark up Tags,Heading Paragraphs, line Breaks. 1.3 HTML5 Tags - Introduction to elements of HTML, Working with Text, Lists, Tables 1.4 Frames 1.5 Hyperlinks, Images and Multimedia 1.6 Forms and other HTML5 controls.	5
2	CO2	1. create different types of style sheets 2. Apply CSS properties to style web pages, 3. Use advanced CSS techniques to build visually appealing and well-structured web layouts. 4. Design and structure responsive web pages using Bootstrap. 5. Integrate JavaScript-based interactive effects	**CSS and Bootstrap** 2.1 Concept of CSS, Creating Style Sheet 2.2 CSS Properties, CSS Styling (Background, Text format, Controlling Fonts), Working with block elements and objects, Lists and Tables 2.3 CSS Id and Class, Box Model (Introduction, Border properties, Padding Properties, Margin properties) 2.4 CSS Advanced: Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector. 2.5 Introduction to Bootstrap, downloading and installing Bootstrap, Introducing the Grid, Offsetting and Nesting, Responsive Features, Utility Classes, and Supported Devices 2.6 Typography in Bootstrap, Styling Tables, Styling Forms, Styling Buttons, Images, icons, and Thumbnails 2.7 Navigation Systems: Tabs, Pills, and Lists, Breadcrumbs and Pagination, Navigation Bar, Making the Navigation Bar Responsive 2.8 JavaScript Effects: Drop-downs, Modal Windows, Tooltips and Popovers, Navigation Aids: Tabs, Collapse, Affix, Carousel	13

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Manipulate web content dynamically 2. Interact with browser-specific objects to create responsive and interactive web pages. 3. Handle browser events effectively 4. Use jQuery to simplify JavaScript programming, 5. Write and integrate PHP scripts to add dynamic content to web pages 6. Design web databases, and establish connectivity between PHP and MySQL 7. Manage web sessions using PHP	**Client-side and Server side-scripting** 3.1 Introduction to JavaScript, Lexical Structure, Types, Values, Variables, Expressions and Operators, Statements 3.2 Objects, Arrays, Functions, Pattern matching with regular expressions 3.3 JavaScript in Web Browsers, The Window object, Scripting Documents, Scripting CSS 3.4 Handling Events jQuery. jQuery Basics, jQuery Getters and Setters 3.5 Altering Document Structure, Handling events with jQuery, Animated Effects, Utility functions, jQuery Selectors and Selection Methods, Extending jQuery with Plug-ins, The jQuery UI Library. 3.6 Introduction to PHP, PHP Tags, Adding Dynamic content, Accessing form variables, Identifiers, user declared variables, Data types, Constants, Operators, Control structures, Conditionals, Iteration constructs, 3.7 Using arrays, string manipulation and regular expressions, reusing code and writing functions 3.8 Designing and creating your web database, Accessing MySQL database from the Web with PHP 3.9 Session Control in PHP	20
4	CO4	1. Explain the Model-View-Controller (MVC) architecture and its significance 2. Apply server-side scripting concepts using the Laravel framework 3. Implement database interactions in Laravel 4. Design user authentication and session management systems within Laravel	**Web Development Framework** 4.1 MVC architecture - Introduction and applications. 4.2 Server side-scripting – Laravel Framework 4.3 Managing Your Project Controllers, Layout, Views, and Other Assets, 4.4 talking to the Database, Model Relations, Scopes 4.5 Other Advanced Features, Integrating Web Forms 4.6 Authenticating and Managing Your Users 4.7 Deploying, Optimizing and Maintaining Your Application	7

VII SUGGESTED SELF LEARNING ACTIVITIES

Microproject to create a responsive website.

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Design static web pages using HTML5	-	-	20%	20%	30%	30%	100%
2	Design layout of web pages using CSS3 and Bootstrap.	-	-	20%	20%	30%	30%	100%
3	Apply the concepts of client-side validation and scripts to static web pages using JavaScript and JQuery and build responsive web pages using front-end framework Bootstrap	-	-	20%	20%	30%	30%	100%
4	Develop a web application using appropriate web development framework	-	-	20%	20%	30%	30%	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Developing Web Applications	2nd Edition	Ralph Moseley, M.T. Savliya	Wiley India	2013
2	Learning PHP, MySQL, JavaScript, CSS	3rd Edition	Robin Nixon	O'Reilly	2014
3	SAMS Teach Yourself Bootstrap in 24 Hours	1st Edition	Jennifer Kyrnin	Pearson Education	2014
4	Laravel 5 Essentials	1st Edition	Martin Bean	Packt Publishing	2015

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	PHP & MySQL in easy Steps	2nd Edition	Mike McGrath	Easy Steps Limited	2012
2	Head First HTML and CSS	2nd Edition	J. Millman & A. Grabel	O'Reilly Media	2012
3	PHP and MySQL Web Development	4th Edition	L. Welling & L. Thomson	Addison-Wesley Professional	2008

Online References

Sr. No	Website Name
A	https://archive.nptel.ac.in/courses/106/106/106106156/, NPTEL Course: Introduction to Modern Application development (IIT Madras);

Program	Third Year B. Tech Electronics & Computer Science (Semester V)	CL	TL	LL	SL	C
Course Name: Software Testing & Quality Assurance	**Course Code:** ECDLOCR15051	2	-	4	2	4
Course Type:	-					
Pre-requisite:	Programming Languages (C++, Java, Python)					

I RATIONALE

In today's software-driven world, developing bug-free code is a challenging task, making software testing essential for ensuring software quality. Testing involves executing a program or application to detect bugs and verify that the product meets its intended requirements. Through this course, students will learn how to identify software defects by applying various types, levels, and methods of testing, supported by an effective test planning approach. The course also includes manual testing techniques.

II COMPETENCY

Apply various types, levels, and methods of software testing on real-world applications through structured teaching-learning experiences.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Analyze software testing methods and strategies to select appropriate techniques for different software development models.
2	Apply manual and automation testing techniques to real-world or commercial software environments
3	Design effective test plans and manage the software testing process
4	Evaluate software quality using testing tools, metrics, and quality standards like ISO 9000 and Six Sigma.

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total	Credits		Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA		ESE	Total		CIAP	ESEP	Total				
								Notional		Duration	MSE	CCE										
				CL	TL	LL		Hours		in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Software Testing & Quality Assurance	-	ECDLOCR15051	2	-	4	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Understand basic architecture and functionality of a web/mobile Application Apply SDLC phases to build the system	Develop Web/Mobile Application (Mini Project)	CO1
2	Analyze code for functionality and possible points of failure	Write a program for any one function of the selected system. Introspect the causes for its failure and write down the possible reasons for its failure.	CO1
3	Translate problem statement into system requirements and a basic design plan	Study the system, requirement specifications and Designing the system	CO1
4	Learn how to create a test plan outlining objectives, scope, strategies, and deliverables	Write the brief test plan	CO3
5	Understand how to choose relevant test cases to validate system behavior in different scenarios	Select the test cases (positive and negative scenarios) for the selected system.	CO3
6	Apply Boundary Value Analysis and Equivalence Class Partitioning to optimize test cases	Design Test cases for the system using boundary value analysis or equivalent class partitioning.	C02
7	Execute test cases without tools- Log and report defects	Manual execution of test cases and prepare defect reports	CO2,CO3
8	Recognize which test cases are best suited for automation	Identify regression scenarios for automation for any one/two test case.	CO2
9	Explore automation tools for functional and regression testing	Study of any testing tool (e.g. Selenium).	CO2
10	Verify that system meets all functional requirements	Functional testing	CO2
11	Implement parameterized tests using various input data	Data driven testing	CO2
12	Understand mutation testing and how it helps evaluate the effectiveness of test cases	Mutation testing for measuring the test case coverage	CO2
13	Evaluate how user-friendly and intuitive the system is	Usability testing	CO2

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Understand the importance of early testing in the development of lifecycles. 2. Differentiate between bugs, errors, faults, and failures. 3. Explain the stages in the testing process: planning, design, execution, and closure. 4. Compare the differences between exhaustive and effective testing. 5. Understand the importance of testing in safety-critical and business-critical systems. 6. Understand the related terms such as test case, test suite, test plan, etc. 7. Identify the entry and exit criteria for each phase. 8. Analyze how testing is planned and executed in Agile vs traditional models.	**Introduction to Software Testing :** 1.1 Introduction, Goals of Software Testing, 1.2 Software Testing Definitions, 1.3 Model for Software Testing, 1.4 Effective Software Testing vs Exhaustive Software Testing, 1.5 Software Failure Case Studies. 1.6 Software Testing Terminology 1.7 Software Testing Life Cycle (STLC). 1.8 Software Testing methodology: V and V testing life cycle, Agile Testing Life Cycle 1.9 Challenges in Agile Testing.	9

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
2	CO2	1. Understand and apply the concept of black box testing BVC. 2. Classify test data using ECT for effective input partitioning. 3. Analyze and apply state table-based testing for systems with defined states and transitions 4. Apply logic coverage criteria (statement, branch, path coverage). 5. Use control flow graphs and cyclomatic complexity for basis path testing. 6. Explain mutation testing and its role in test effectiveness. 7. Apply static testing techniques like code reviews and inspections. 8. Define various validation levels and when they are used in the SDLC. 9. Differentiate between Progressive and Regressive Regression Testing and their applications. 10. Understand Object-Oriented Testing (OOT) fundamentals and challenges. 11. Apply OOT techniques like Class Testing, Interaction Testing, and State-Based Testing. 12. Understand the need for automation in testing large/complex systems. 13. Evaluate factors for tool selection including ease of use, integration, and community support.	**A Black Box testing:** 2.1 boundary value analysis 2.2 equivalence class testing 2.3 state table-based testing, error guessing. **B White box Testing Techniques:** 2.4 need, logic coverage criteria basis path testing 2.5 loop testing data flow testing 2.6 mutation testing 2.7 Static Testing. 2.8 Validation Activities: Unit validation, Integration, Function, System 2.9 Acceptance Testing, Regression testing (Progressive v/s Regressive Testing). 2.10 Testing Object-Oriented Software: OOT Basics 2.11 Object-oriented Testing. 2.12 Automation and Testing Tools: need, categorization 2.13 selection and cost in testing tool, Guidelines for testing tools	12

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Understand how testing teams are structured in an organization and their roles. 2. Identify responsibilities, required skills, and inter-group communication strategies. 3. Learn to create a test plan including objectives, scope, resources, and schedule. 4. Understand how to design test cases, test data, and write clear test specifications. 5. Apply techniques to minimize the size of a test suite while maintaining adequate test coverage	**Design and Managing the Test Process** 3.1 Test Management: test organization, structure and of testing group 3.2 Test planning, detailed test design and test specification. 3.3 Software Metrics: need, definition and classification of software matrices. 3.4 Efficient Test Suite Management: minimizing the test suite and its benefits.	5
4	CO4	1. Understand the concept of SQM and its role in ensuring software meets user and business expectations. 2. Identify and explain McCall's 11 quality factors and how they guide software evaluation. 3. Understand the ISO 9000:2000 standard and its application in software process quality management. 4. Apply Six Sigma concepts to measure, analyze, and improve software quality and reduce defects.	**Quality Management** 4.1 Software Quality Management 4.2 McCall's quality factors and Criteria. 4.3 ISO9000:2000 4.4 SIX Sigma.	4

VII SUGGESTED SELF LEARNING ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related co-curricular activities eg.documentation,Presentation and mooc's courses etc. To explore various software testing methodologies (manual and automated) and implement them using different open source test-

ing tools(Selenium, Postman, JUnit etc) to understand their effectiveness, features, and applications in real-world scenarios.

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Analyze software testing methods and strategies to select appropriate techniques for different software development models.	-	-	50%	50%	-	-	100%
2	Apply manual and automation testing techniques to real-world or commercial software environments	-	-	50%	50%	-	-	100%
3	Design effective test plans and manage the software testing process	-	-	-	-	100%	-	100%
4	Evaluate software quality using testing tools, metrics, and quality standards like ISO 9000 and Six Sigma.	-	-	100%	-	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Software Testing Principles and Practices	2nd Edition	Naresh Chauhan	Oxford Higher Education	2016
2	Effective Methods for Software Testing	3rd Edition	Willam E. Perry	Wiley Publication	2006

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Software Testing Concepts and Tools	1st Edition	Nageswara Rao Pusuluri	Dreamtech press	2006
2	Software Testing and quality assurance theory and practice	1st Edition	Kshirasagar Naik, Priyadarshi Tripathy	Wiley Publication	2007

Reference Books

Online References

Sr. No	Website Name
A	https://www.selenium.dev, Selenium tool
B	https://www.youtube.com/live/Lm75Zd12-W8?si=3en08mh6_CMmI1UJ, Automation Testing Full course -Edureka
C	https://www.youtube.com/live/sO8eGL6SFsA?si=k9t_jxC7Z57Hkw43, Software testing Tutorial Intellipat

Program	Third Year B.Tech Electronics & Computer Science(Semester V)	CL	TL	LL	SL	C
Course Name: System Verification With System Verilog	**Course Code:** ECDLOCR15052	2	-	4	2	4
Course Type:	-					
Pre-requisite:	C Programming					

I RATIONALE

Learning System Verilog for system verification provides a robust, standardized, and scalable solution that meets the demands of modern digital design. It bridges the gap between hardware design and verification and forms the backbone of contemporary verification methodologies such as UVM.

II COMPETENCY

Apply System Verilog language and associated methodologies such as UVM to plan, develop, implement, and maintain verification environments for digital systems.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Demonstrate programmable devices and verification methodologies.
2	Examine new constructs in System Verilog and summarize ASIC verification techniques such as Randomization, assertions, coverage.
3	Create layered test benches for digital designs in system Verilog.
4	Carry out verification of design successfully using simulators.

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total	Credits	Theory						Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA					CIAP	ESEP	Total				
											MSE	CCE	ESE	Total								
				CL	TL	LL		Notional Hours		Duration in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	System Verification With System Verilog	-	ECDLOCR15052	2	-	4	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Apply Verilog modeling styles and simulate 4:1 MUX using Verilog.	Design and simulate 4:1 MUX using Verilog modeling styles.	CO 1
2	Apply Verilog modeling styles and simulate D flip flop and 4 bit counter using Verilog.	Design and simulate D flip flop and 4 bit counter using Verilog modeling styles.	CO 2
3	Apply test bench concepts and simulate ALU using Verilog.	Create a test plan and self-checking test-bench for the ALU.	CO 2
4	Apply System Verilog modeling styles and Create dynamic arrays, associative arrays, and queues.	Create dynamic arrays, associative arrays, and queues using System Verilog.	CO 2
5	Apply System Verilog modeling styles and Write test bench using dynamic arrays, associative arrays with System Verilog to test a synchronous 8-bit x64K (512kBit) RAM.	Write test bench using dynamic arrays, associative arrays with System Verilog to test a synchronous 8-bit x64K (512kBit) RAM.	CO 2
6	Create an Interface for a Memory Design.	Create an Interface for a Memory Design. Use Modports to assign direction to signal.	CO 3
7	Apply Verilog modeling styles in fork join/join any, join none and implement parallel processes.	Implement of parallel processes using Fork Join/ join any/ join none statement.	CO 3
8	Apply thread concepts in system verilog and create IPCs like events, mailbox and semaphores.	Create IPCs like events, mailbox and semaphores to interact between threads.	CO 4

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO 1	1. Describe Fundamentals of CPLD, FPGA, ASIC, SoC,SiP, MCM and SoP. 2. Apply test bench concepts into system verification process.	**Programmable Devices and Verification Basics** 1.1 Programmable Devices: Different types of Integrated Circuits- CPLD, FPGA,ASIC, SoC (System -on- Chip), SiP (System-in-Package), MCM (Multi-Chip Module), SoP (System-on-Package), Choices based on application and cost, Architecture of FPGA, CPLD (Xilinx and Altera family devices), Difference between ASIC, FPGA and CPLD, ASIC flow and overview of types of tools used in each stage of life cycle. 1.2 Verification Basics: Introduction, Verification Process, Verification Plan, Verification Methodology options, Basic Test bench Functionality, Directed Testing, Constrained-Random Stimulus, Functional Coverage, Test bench Components, Layered Test bench, Technology challenges test, Verification.	8
2	CO 2	1. Examine data types, procedural statements concepts in system design. 2. Connect test bench and design in system verilog.	**Data types, Procedural statements, Connecting the Test bench and Design** 2.1 Data Types: Built-in Data Types, Logic Data type, Fixed-Size Arrays (Packed and Unpacked) 2.2 DProcedural statements: Procedural Statements, Tasks, Functions, and Void Functions, routine arguments, returning from a routine, Time values. 2.3 Connecting the Test bench and Design: Separating the test-bench and design, The Interface.	8

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO 3	1. Analyze randomization techniques to describe inter process communication. 2. Apply threads and IPC techniques to describe inter process communication	**Randomization and Inter-process Communication** 3.1 Randomization: Randomization in system Verilog, Constraint details, Solution probabilities, controlling multiple constraint blocks, Valid constraints, In-line constraints, The pre-randomize and post- randomize functions. 3.2 Threads and Inter-process Communication: Working with threads, disabling threads, inter- process communication, Events, Semaphores, Mailboxes, building a test-bench with threads and IPC.	8
4	CO 4	1. Apply concepts of assertions in system verilog. 2. Explain coverage methods to analyze data.	**System Verilog Assertions and Functional Coverage** 4.1 System Verilog Assertions: Types of Assertions and examples, Immediate Assertions, Concurrent Assertions, built-in methods (*rose*,fell, *stable*,past). 4.2 Functional Coverage: Coverage Types, Functional Coverage Strategies, Simple Functional coverage Example, anatomy of a cover group, triggering a cover group, data sampling, cross coverage, generic cover groups, Coverage Options, Parameterized Cover Groups, Analyzing Coverage Data.	6

VII SUGGESTED SELF LEARNING ACTIVITIES

1. Perform micro project on system design using system verilog. 2. Prepare a survey article/review paper on verilog/system verilog in programmable devices.

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Demonstrate programmable devices and verification methodologies.	-	-	50%	50%	-	-	100%
2	Examine new constructs in System Verilog and summarize ASIC verification techniques such as Randomization, assertions, coverage.	-	-	40%	60%	-	-	100%
3	Create layered test benches for digital designs in system Verilog.	-	-	70%	30%	-	-	100%
4	Carry out verification of design successfully using simulators.	-	-	40%	60%	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	System Verilog for Verification: A guide to learning the test bench language features	2rd Edition	Chris Spear	springer	2010
2	"System Verilog for Design: A guide to using system verilog for hardware design and modeling"	2 rd Edition	Stuart Sutherland, Simon Davidmann, and Peter Flake	Springer	2004

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Writing Test benches Using System Verilog	3rd Edition	Janick Bergeron	springer	2006
2	Verilog HDL: A guide to Digital Design and Synthesis	2nd Edition	Samir Palnitkar	Pearson	2001

Online References

Sr. No	Website Name
A	www.xilinx.com/ug440-xilinx-power-estimator ,Spartan and Virtex family user manuals from Xilinx
B	https://www.hdlworks.com/hdl_corner/verilog_ref/, Verilog Language Reference manual
C	www.edtwiki.org/systemverilog_3.1_manual ,System verilog reference manual
D	https://www.asic-world.com/systemveriloghttps://verificationacademy.com, (Siemens)

Program	Third Year B Tech Electronics & Computer Science (SemesterV)	CL	TL	LL	SL	C
Course Name: Signals and Systems	**Course Code:** ECDLOCR15053	2	-	4	2	4
Course Type:	-					
Pre-requisite:	Engineering Mathematics, Laplace Transform, Z Transform, Fourier Series, Fourier Transform					

I RATIONALE

The subject "Signals and Systems" serves as a fundamental cornerstone in the fields of Electrical, Electronics, Instrumentation, and Communication Engineering, as well as Computer Science and AI domains. It provides the essential tools and concepts required to analyze, design, and implement systems that process signals in both time and frequency domains.

Understanding signals, how they are generated, modified, and interpreted is crucial in real-world applications such as audio processing, image filtering, communication systems, biomedical signal analysis, radar, and control systems.

II COMPETENCY

Apply mathematical and analytical techniques to model, analyze, and interpret continuous-time and discrete-time signals and systems in both time and frequency domains, using tools such as Laplace and Z-Transforms, to evaluate system behavior, stability, and response for real-world engineering applications.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Select continuous time and discrete time Signals and Systems.
2	Apply various transforms for time domain to frequency domain conversion.
3	Apply frequency domain techniques for analysis of LTI systems.
4	Apply frequency domain techniques for analysis of continuous and discrete signals.

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme					Credits	Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total			Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code					Notional		Exam	CIA		ESE	Total		CIAP	ESEP	Total				
				CL	TL	LL		Hours		Duration in Hrs	MSE Max	CCE Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Signals and Systems	-	ECDLOCR15053	2	-	4	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Any 12)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Understand and navigate the MATLAB environment to perform basic mathematical and signal processing operations.	Introduction to MATLAB	CO1
2	Analyze and interpret the time-domain behavior of a first-order system to understand transient and steady-state characteristics.	Basic plotting of signals (2D and 3D plots)	CO1
3	Generate and distinguish between continuous-time and discrete-time signals through appropriate plotting techniques.	Basic plotting of continuous and discrete time signals	CO1
4	Develop continuous-time standard signals (e.g., sinusoidal, exponential, step) using MATLAB scripting.	Generation of continuous time signals	CO1
5	Apply arithmetic operations (addition, multiplication, scaling) to continuous-time signals and interpret the results.	Different operations on continuous time signals	CO2
6	Implement time shifting, time scaling, and amplitude scaling on signals and analyze their effects.	Time and Amplitude transformations	CO2
7	Use MATLAB to compute linear convolution and analyze system response to input signals.	Convolution of given signals.	CO3
8	Calculate and interpret autocorrelation and cross-correlation to analyze signal similarity and delay.	Autocorrelation and Cross-correlation.	CO3
9	Evaluate system properties such as linearity, time invariance, and causality using signal tests in MATLAB.	Properties of systems.	CO3
10	Compute Laplace or Z-transforms of signals using MATLAB and interpret their frequency-domain behavior.	Calculating transforms using MATLAB.	CO4
11	Derive and analyze the impulse and step response of systems to characterize system dynamics.	Impulse response and Step response of a given system.	CO4
12	Plot and analyze pole-zero and Bode diagrams to understand system stability and frequency response.	Pole-zero diagram and Bode diagram	CO3
13	Use SIMULINK to simulate systems and verify properties like linearity and non-linearity visually.	Checking Linearity/Non-Linearity of a system using SIMULINK.	CO4

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Identify and classify continuous-time (CT) and discrete-time (DT) signals based on their characteristics. 2. Analyze the orthogonality of signals and apply it in signal decomposition. 3. Perform basic arithmetic and time-domain operations such as shifting, scaling, and reversal on CT and DT signals. 4. Explain and demonstrate the process of signal sampling, reconstruction, and identify aliasing effects in signal conversion.	**Continuous and Discrete Time Signals** 1.1 Mathematical Representation and Classification of CT and DT signals, Orthogonality of signals 1.2 Arithmetic operations on the signals, Time Shifting, Time Scaling, Time Reversal of signals 1.3 Sampling and Reconstruction, Aliasing effect .	6
2	CO2	1. Classify and represent CT and DT systems using mathematical models. 2. Determine system properties such as linearity, time-invariance, causality, and stability. 3. Compute system output using convolution integral (for CT) and convolution sum (for DT). 4. Apply correlation techniques to compare signals and analyze system behavior.	**Continuous and Discrete Systems** 2.1 Mathematical Representation and classification of CT and DT systems 2.2 Properties of LTI systems, impulse, and step response. 2.3 Use of convolution integral, convolution sum, and correlation for analysis of LTI systems 2.4 Properties of convolution integral and convolution sum	6

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Understand and apply the concept of complex frequency and Region of Convergence (ROC) for different system types (causal, non-causal, anti-causal). 2. Perform Laplace transforms and inverse transforms for system analysis. 3. Analyze system characteristics such as impulse and step response, causality, and stability using Laplace domain techniques. 4. Interpret the pole-zero plot to infer system behavior in frequency domain.	**Frequency Domain Analysis of Continuous Time System using Laplace Transform** 3.1 Concept of Complex frequency, Region of Convergence for Causal, Non-causal, and Anti-causal systems, Poles, and Zero of the transfer function 3.2 Unilateral Laplace Transform 3.3 Analysis and characterization of LTI system using Laplace Transform: Impulse and Step Response, Causality, Stability, Stability of Causal system	9
4	CO4	1. Understand and explain the need and significance of Z Transform for discrete systems and its relationship with Laplace transform. 2. Compute and analyze Z transforms of standard signals, determine ROC, poles, and zeros. 3. Evaluate the behavior and characteristics of LTI systems (impulse response, step response, stability) using Z Transform. 4. Implement various forms of system realization (Direct, Canonic, Cascade, and Parallel) using transfer functions.	**Frequency Domain Analysis of Discrete Time System using Z Transform** 4.1 Need for Z transform, definition, properties of unilateral and bilateral Z Transform, mapping with s plane, relationship with Laplace transform 4.2 Z transform of standard signals, ROC, poles, and zeros of the transfer function, Inverse Z transform 4.3 Analysis and characterization of the LTI system using Z transform, impulse and step response, causality, stability, stability of the causal system 4.4 System Realization-Direct, Direct Canonic, Cascade and Parallel forms	9

VII SUGGESTED SELF LEARNING ACTIVITIES

Case study/Research Paper survey/Microproject/Literature Review Report and Presentation

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Select continuous time and discrete time Signals and Systems. -	-	20%	60%	20%	-	-	100%
2	Apply various transforms for time domain to frequency domain conversion.	-	20%	60%	20%	-	-	100%
3	Apply frequency domain techniques for analysis of LTI systems.	-	20%	60%	20%	-	-	100%
4	Apply frequency domain techniques for analysis of continuous and discrete signals.	-	20%	60%	20%	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Signals and Systems	1st Edition	Tarun Kumar Rawat	Oxford University Press	2016
2	Signals and Systems	2nd Edition	A. NagoorKani	Tata McGraw-Hill Education	2022

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Digital Signal Processing	4th Edition	John Proakis and Dimitris Monolakis	Pearson Publication	2007
2	Signals and Systems	2nd Edition	Alan V. Oppenheim, AlanS. Willsky, and S.Hamid Nawab	John Wiley and Sons	2013
3	Linear Systems and Signals	3rd Edition	B. P. Lathi	Oxford University Press	2018

Online References

Sr. No	Website Name
A	https://ocw.mit.edu/courses/res-6-007-signals-and-systems-spring-2011/
B	https://vemu.org/uploads/lecture_notes/04_03_2021_1450719340.pdf

Program	Third Year B.Tech Electronics and Computer Science (Semester - V)	CL	TL	LL	SL	C
Course Name: Sensor technologies and applications	**Course Code:** ECDLOCR15054	2	-	4	2	4
Course Type:						
Pre-requisite:	Basic Electrical and Electronic Engineering, Analog and Digital Electronics, Engineering Physics, Mathematics for Engineers.					

I RATIONALE

The course on Sensor Technology equips students with the knowledge and analytical skills necessary to understand and apply different types of sensors used in modern engineering systems. The course introduces students to the principles, materials, and standards associated with sensor technology, enabling them to design, troubleshoot, and apply sensors in diverse domains such as industrial automation, biomedical engineering, consumer electronics, and smart systems.

II COMPETENCY

Evaluate sensor principles and characteristics, Designing and interfacing sensor circuits, Applying sensors in diverse real-life scenarios.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Analyze the working principles and characteristics of various sensors
2	Analyze design approaches using various sensors for specific technological applications.
3	Examine different sensor materials and technologies to determine their suitability in design.
4	Analyze sensor-based problems and apply troubleshooting techniques in real-world scenarios.

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme												
				Actual Contact Hrs/Week			SL	Total Notional Hours	Credits		Theory					Practical				SLA		Total
										Exam Duration	CIA		ESE	Total		CIAP	ESEP	Total			
Sr. No.	Course Title	Course Type	Course Code	CL	TL	LL				in Hrs	MSE Max	CCE Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Sensor technologies and applications	–	ECDLOCR15054	2	-	4	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	LO1: Analyze the operational characteristics of various temperature sensors.	Characteristics of Temperature Sensors	CO1
2	LO2: Examine the working principles and characteristics of optical sensors.	Characteristics of Optical Sensors	CO1, CO2
3	LO3: Design and evaluate I–V and V–I converters using OpAmps.	I to V and V to I Converter	CO2, CO4
4	LO4: Implement and analyze a frequency-to-voltage converter.	Frequency to Voltage Converter using OpAmp	CO2, CO4
5	LO5: Construct and analyze amplifier circuits using OpAmp.	Inverting and Non-inverting Amplifier using OpAmp	CO2, CO4
6	LO6: Investigate the construction and behavior of an LVDT sensor.	LVDT Sensor Construction and Characteristics	CO1, CO3
7	LO7: Design and evaluate an instrumentation amplifier for sensor applications.	Instrumentation Amplifier Design	CO2, CO3, CO4
8	LO8: Analyze and design analog filters.	Instrumentation Amplifier Design	CO2, CO4
9	LO9: Simulate and analyze digital filters using software tools.	Filter Design (Digital Simulation)	CO2, CO4
10	LO10: Analyze the sensor-based technology in household appliances.	Case Study on Any Household Appliance	CO1, CO2
11	LO11: Examine the operation of a 4-20mA loop for sensor signal transmission.	4-20mA Current Loop	CO1, CO4
12	LO12: Implement and analyze A/D conversion for sensor data.	Interfacing with Real World using A/D Converters	CO2, CO4
13	LO13: Implement and analyze D/A conversion for actuators.	Interfacing with Real World using D/A Converters	CO2, CO4
14	LO14: Simulate and interpret the behavior of micro-sensors.	Simulations of Micro-sensors	CO1, CO3
15	LO15: Simulate and analyze micro-actuators for MEMS applications.	Simulations of Micro-actuators (e.g., micro-heater/micro-motors)	CO1, CO3

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. LO1.1: Classify various sensors based on physical, mechanical, electrical, and chemical principles. 2. LO1.2: Analyze sensor characteristics and describe the underlying physical principles such as capacitance, induction, resistance, piezoelectric and Hall effects, etc.	**Sensors Fundamentals and Characteristics Sensors, Signals and Systems** 1.1 Sensor Classification–Physical, Mechanical, Electrical, Chemical, electrochemical 1.2 Sensor Characteristics, Physical Principles of Sensing Electric Charges, Fields, and Potentials; Capacitance; Magnetism; Induction; Resistance; Piezoelectric Effect; Hall Effect; Temperature and Thermal Properties of Material; Heat Transfer; Light; Dynamic Models of Sensor Elements	5
2	CO2	1. LO2.1: Examine input characteristics of interface circuits including amplifiers and excitation systems. 2. LO2.2: Apply knowledge of A/D conversion and data transmission in sensor interfacing. 3. LO2.3: Compare the working principles and applications of motion, position, force, pressure, and tactile sensors. 4. LO2.4: Correlate the transduction principles of various sensors like temperature, biosensors, gas, and proximity sensors with measurable electrical outputs. 5. LO2.5: Analyze real-world applications of sensors in automotive, manufacturing, and consumer digital devices.	**Interface Electronic Circuits and Its Applications** 2.1 Input Characteristics of Interface Circuits, Amplifiers, Excitation Circuits 2.2 Analog to Digital Converters, Direct Digitization and Processing, Bridge Circuits, Data Transmission, Batteries for Low Power Sensors 2.3 Area Occupancy and Motion Detectors; Position, Displacement, and Level; Velocity and Acceleration; Force, Strain, and Tactile Sensors; Pressure Sensors 2.4 Temperature Sensors; Biosensors, Gas sensors, proximity sensor. (Correlation of output with the parameter being measured in engineering terms): Only Working principle of each type of sensors and transduction action (for example: detection of change in temperature and conversion to electrical quantity say resistance and corresponding correlation) 2.5 Case study of Applications of sensors in Automotive, Manufacturing plants, digital devices.	9

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. LO3.1: Describe the structure, material, and working principles of MEMS-based sensors like accelerometers and gyroscopes. 2. LO3.2: Analyze the construction and sensing mechanism of metal-oxide semiconductor-based sensors for gas, biomedical, and chemical detection.	**Sensor Materials and Technologies** 3.1 MEMS-cantilever based sensors and their types such as, accelerometer, gyroscopes: Structure, material used (polysilicon, Silicon etc), working principle, applications. 3.2 Metal oxide semiconductor (nano-particles) based sensors such as gas sensors, biomedical sensors, chemical sensors (Structure, material used, working principle, applications)	7
4	CO4	1. LO4.1: Explain the concept and implementation of a 4–20 mA current loop in industrial applications. 2. LO4.2: Differentiate between single sensors and sensor arrays; discuss applications of smart sensors and electronic nose systems. 3. Describe industrial communication standards such as HART, Profibus, and CANbus used for sensor interfacing. 4. Summarize the key provisions of standards like IEC 60601-1-1, ISA S82.01, and NEMA relevant to sensor safety and compliance.	**Smart Sensors and Industrial standards for the sensors and its calibration** 4.1 4-20 mA Current Loop 4.2 Types of smart Sensors, Limitations of single sensor and applicability of Array based sensor technology, Electronic - Nose sensors 4.3 HART, Industrial buses such as Profibus, CANbus, etc. 4.4 Basic knowledge about IEC 60601-1-1: Medical Electrical Equipment – Part 1- 1, ISA S82.01, NEMA standards	9

VII SUGGESTED SELF LEARNING ACTIVITIES

We will make logbook for activities conducted on - micro project, mini project, case study, simulation project, arduino / ESP32 / Raspberry Pi and sensor based system.

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Sensors Fundamentals and Characteristics Sensors, Signals and Systems	-	-	50 %	50 %	-	-	100 %
2	Interface Electronic Circuits and Its Applications	-	-	50 %	50 %	-	-	100 %
3	Sensor Materials and Technologies	-	-	50 %	50 %	-	-	100 %
4	Smart Sensors and Industrial standards for the sensors and its calibration	-	-	50 %	50 %	-	-	100 %

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Handbook of Modern Sensors: Physics, Designs, and Applications	4th Edition	Jacob Fraden	Springer	2015
2	Sensors and Transducers	2nd Edition	D. Patranabis	PHI Learning Pvt. Ltd., New Delhi	2003
3	Mechatronics	2nd Edition	Ganesh S. Hegde	University Science Press (Laxmi Publications)	2010
4	Process Control Systems and Instrumentation	New Edition	Terry Bartelt	Delmar Cengage Learning, India Edition	2006

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Metal Oxide Nanostructures as Gas Sensing Devices	1st Edition	G. Eranna	CRC Press	2013
2	ISA S82.01 - Safety Standard for Electrical and Electronic Equipment	(Standard Document)	ISA (Instrument Society of America)	ISA (International Society of Automation)	1994

Online References

Sr. No	Website Name
1	https://nptel.ac.in,Offers free online video lectures and certifications from IITs on topics such as sensors, instrumentation, and MEMS technologies
2	https://www.ti.com,Texas Instruments' site offers sensor datasheets, application notes, and design resources helpful for understanding industrial sensor implementations.
3	https://www.analog.com,Analog Devices provides comprehensive sensor product guides, signal conditioning techniques, and interface circuit designs.
4	https://www.instructables.com,Hands-on projects related to sensor interfacing with Arduino, Raspberry Pi, and other platforms—useful for practical self-learning.

Program	Third Year B Tech Electronics & Computer Science (SemesterV)	CL	TL	LL	SL	C
Course Name: Control System and Instrumentation	**Course Code:** ECDLOCR15055	2	-	4	2	4
Course Type:	-					
Pre-requisite:	Basic Electrical & Electronics Engineering, Engineering Mathematics I & II (including DE and Laplace)					

I RATIONALE

The Controls and Instrumentation course is fundamental for engineering students, especially in disciplines such as Electrical, Electronics, Instrumentation, and Mechanical Engineering. It builds a comprehensive understanding of how systems behave, how they can be modeled mathematically, and how they can be monitored and controlled effectively in real-world industrial and automation contexts. This course Control System and Instrumentation is designed to provide students with comprehensive knowledge and practical understanding of modern control theory and industrial instrumentation systems. With increasing complexity in engineering systems and the need for precision, automation, and reliability, understanding advanced control mechanisms and instrumentation has become critical across multiple industries including electrical, electronics, automation, robotics, and process industries.

II COMPETENCY

Model dynamic systems using transfer functions, block diagrams, and signal flow graphs, and analyze their time responses.

Assess system stability in both time and frequency domains using classical methods like Routh-Hurwitz, Root Locus, Bode, and Nyquist plots.

Select appropriate sensors and transducers based on their operating principles and application requirements.

Apply concepts of data acquisition systems, data loggers, and SCADA for monitoring and controlling industrial processes.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Analyze control systems using transfer functions, block diagram reduction, and signal flow graphs, time-domain responses of first and second-order systems.
2	Assess the stability of control systems using Routh-Hurwitz, Root Locus, Bode, and Nyquist techniques.
3	Select appropriate sensors and transducers for measuring physical quantities
4	Describe the architecture and functions of DAS, data loggers, and SCADA systems for industrial data acquisition and control.

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme													
				Actual Contact Hrs/Week			SL	Total	Credits		Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA		ESE	Total		CIAP	ESEP	Total				
								Notional		Duration	MSE	CCE										
				CL	TL	LL		Hours		in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Control System and Instrumentation	-	ECDLOCR15055	2	-	4	2	120	4	2	-	50	-	50	20	25	25	50	20	25	10	125

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 12)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Understand the interface, features, and basic functionalities of MATLAB and Simulink control system toolboxes for system modeling and simulation.	Familiarization with the Matlab control system toolbox, and Matlab/Simulink toolbox.	CO1
2	Analyze and interpret the time-domain behavior of a first-order system to understand transient and steady-state characteristics.	Determination of step and impulse response for a first-order unity feedback system	CO1
3	Evaluate the time-domain response of a second-order system and identify the impact of damping and natural frequency on system dynamics.	Determination of step and Impulse response for a second-order unity feedback system	CO1
4	Investigate how system type affects steady-state error and dynamic response using standard inputs like step and impulse.	Determination of step and impulse response for type '0', type'1',type'2'systems	CO1
5	Generate and analyze Bode plots to extract gain margin, phase margin, bandwidth, and other frequency response specifications.	Determination of bode plot using MATLAB control system toolbox for2ndorder system and obtain controller specification parameters.	CO2
6	Construct and interpret root locus plots to analyze system stability and design control parameters.	Determination of root locus plot using MATLAB control system toolbox for 2nd order system and obtain controller specification parameters.	CO2
7	Evaluate the influence of Proportional-Integral (PI) and Proportional-Derivative (PD) controllers on system time-domain performance such as overshoot, settling time, and steady-state error.	Study the effect of PI and PD controller on system performance	CO2
8	Understand the working principle of a Linear Variable Differential Transformer (LVDT) and apply it to accurately measure linear displacement.	Displacement measurement using LVDT.	CO3
9	Demonstrate the operation of strain gauges and use them to measure applied pressure through strain-resistance relationships.	Pressure Measurement using Strain Gauge	CO3
10	Learn the thermoelectric principle behind thermocouples and apply them for temperature sensing across a measurable range.	Temperature measurement using Thermocouple.	CO3

(Continued on next page)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
11	Develop and simulate basic ladder logic programs for control applications using Programmable Logic Controllers (PLC).	PLC programming with Ladder Diagram	CO4
12	Understand SCADA architecture and its practical application in industrial process monitoring and control through software demonstration.	Demonstration of the SCADA system using open-Source software	CO4
13	Explain the working of capacitive displacement sensors and implement them in displacement measurement tasks.	Displacement measurement using capacitive transducer	CO3
14	Analyze and interpret the response of a Resistance Temperature Detector (RTD) sensor under different temperature conditions.	Characterize the temperature sensor(RTD)	CO3
15	Use ladder logic in PLCs to implement basic digital logic operations such as AND, OR, NOT, NAND, and NOR gates.	Implementation of Logic Gates using PLC	CO4

VI THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Understand the basic concepts, types, and components of control systems (open-loop and closed-loop systems). 2. Derive and interpret mathematical models of physical systems using transfer functions. 3. Analyze control systems using block diagram algebra and perform block diagram reduction to simplify complex systems. 4. Use signal flow graphs and Mason's Gain Formula to determine overall system transfer functions. 5. Understand and apply standard test signals (step, impulse, ramp, parabolic) for time-domain analysis of control systems. 6. Determine and analyze the time response of first-order and second-order systems, including transient and steady-state behaviors. 7. Calculate steady-state errors and error constants (position, velocity, and acceleration error constants) for different types of inputs and system types. 8. Correlate time response parameters (rise time, settling time, peak time, overshoot) with system performance and stability.	**Introduction to Control Systems, Mathematical Models and Time response analysis** 1.1 Introduction to control systems: Mathematical models: Transfer functions, block diagram algebra, block diagram reduction, signal flow graphs. 1.2 Time response analysis: Time response analysis: standard test signals, time response of first and second order systems, steady state errors and error constants.	8

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
2	CO2	1. Understand the concept of stability in control systems and identify the types of stability (absolute, relative, bounded input-bounded output). 2. Apply necessary and sufficient conditions for stability using characteristic equations. 3. Analyze system stability using the Hurwitz stability criterion. 4. Apply the Routh-Hurwitz criterion to determine system stability and assess the impact of parameter variations. 5. Perform relative stability analysis to determine how stable a system is beyond basic stability. 6. Construct and interpret root locus plots to assess how pole locations change with varying system parameters and how this affects system stability and performance. 7. Understand the frequency response of a system and how it relates to time-domain behavior. 8. Correlate frequency-domain characteristics (gain margin, phase margin, bandwidth) with system stability and performance. 9. Construct and analyze Bode plots to evaluate system behavior and design control parameters. 10. Apply the Nyquist stability criterion and construct Nyquist plots to determine closed-loop system stability.	**Stability analysis in time domain and Frequency domain** 2.1 Stability in time domain: The concept of stability, necessary conditions for stability, Hurwitz stability criterion, Routh stability criterion, relative stability analysis. Stability analysis using root locus technique. 2.2 Stability Analysis in frequency domain: Introduction to frequency response analysis, correlation between time and frequency domain. Stability analysis using Bode plots. Nyquist stability criterion and stability analysis using Nyquist plot.	8

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Understand the basic concepts, definitions, and differences between sensors and transducers. 2. Identify and classify various types of sensors and transducers based on physical quantity measured (e.g., temperature, pressure, displacement). 3. Explain the principles of operation for different types of transducers. 4. Evaluate and apply the criteria for selecting transducers based on factors like range, sensitivity, linearity, environmental conditions, and application requirements. 5. Describe the working principle and applications of displacement transducers such as potentiometers and LVDTs. 6. Explain the operation and use of strain gauges and pressure gauges for pressure measurement. 7. Analyze the behavior and signal characteristics of these transducers under different physical inputs. 8. Explain the working principles, characteristics, and applications of common temperature transducers: • RTD (Resistance Temperature Detector) • Thermocouple • Thermistor 9. Compare temperature transducers based on range, sensitivity, response time, and accuracy. 10. Select suitable temperature sensors for specific measurement environments and system requirements.	**Sensors and Transducers** 3.1 Introduction to sensors and transducers: Various types of sensors. Various types of transducers and their principle of operation. Selection criteria of transducers. 3.2 Displacement and pressure transducers: potentiometers, pressure gauges, Linear variable differential transducer (LVDT), strain gauges. 3.3 Temperature transducers: working principle, ranges and applications of resistance temperature detectors (RTD), thermocouple and thermistor temperature transducers.	8

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
4	CO4	1. Understand the structure and function of instrumentation systems and their role in monitoring and control processes. 2. Explain the concept, components, and working of a Data Acquisition System (DAS) and its role in intelligent instrumentation systems. 3. Identify different types of data loggers, their working principles, and applications in industrial and laboratory environments. 4. Describe the architecture and communication protocols of SCADA (Supervisory Control and Data Acquisition) systems. 5. Differentiate between types of SCADA systems, their applications in automation, and the significance of open SCADA protocols (e.g., Modbus, DNP3, OPC).	**Signal conditioning DAS, Data logger and SCADA** 4.1 DAS, DATA Logger and SCADA: Introduction to instrumentation systems, Data acquisition system (DAS), use of DAS in Intelligent instrumentation system. Data logger, its types and applications. SCADA communication architecture, types, applications, open SCADA protocols.	6

VII SUGGESTED SELF LEARNING ACTIVITIES

Case study/Research Paper survey/Microproject/Literature Review Report and Presentation

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Analyze control systems using transfer functions, block diagram reduction, and signal flow graphs, time-domain responses of first and second-order systems. -	-	20%	60%	20%	-	-	100%
2	Assess the stability of control systems using Routh-Hurwitz, Root Locus, Bode, and Nyquist techniques.	-	20%	60%	20%	-	-	100%
3	Select appropriate sensors and transducers for measuring physical quantities.	-	20%	60%	20%	-	-	100%
4	Describe the architecture and functions of DAS, data loggers, and SCADA systems for industrial data acquisition and control.	-	20%	60%	20%	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Control System Engineering	6th Edition	I. J. Nagrath and M. Gopal	New Age International Publishers	2017
2	Linear Control Systems	11th Edition	B. S. Manke	Khanna Publishers, New Delhi	2012
3	Principle of Industrial Instrumentation	3rd Edition	D. Patranabis	Tata McGraw Hill	2012-2013
4	Electrical & Electronic Measurement & Instrumentation	19th Edition	A. K. Sawhney	DRS India	2014
5	Electronic Instrumentation	4th Edition	H. S. Kalsi	Tata McGraw Hill	2019
6	Control Systems: Principles and Design	3rd Edition	M. Gopal	Tata McGraw Hill	2008
7	Modern Control Systems	11th Edition	Richard Dorf, Robert Bishop	Pearson Education	2008

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Modern Control Engineering	5th Edition	K. Ogata	PHI, New Delhi	2021
2	Control System Engineering	8th Edition	Norman S. Nise	John Wiley and Sons	2024
3	Automatic Control Systems	10th Edition	B. C. Kuo	PHI, New Delhi	2017
4	Instrumentation Devices and Systems	2nd Edition	C. S. Rangan, G. R. Sharma, V. S. Mani	Tata McGraw-Hill	2001
5	Modern Electronic Instrumentation & Measuring Techniques	1st Edition	Helfrick & Cooper	PHI	2016

Online References

Sr. No	Website Name
A	https://onlinecourses.nptel.ac.in/noc25_de14/preview, NPTEL Course: Controls System
B	https://l1nq.com/sXUJv, Book Science Direct

Program	Third year B Tech Electronics & Computer Science(Semester V)	CL	TL	LL	SL	C
Course Name: Skill-based Lab IV- Linux Server Administration Lab	**Course Code:** ECLR1506	-	-	4	-	2
Course Type:	-					
Pre-requisite:	Knowledge of Computer operating system					

I RATIONALE

Learning a Linux server administration course provides valuable skills for managing and maintaining Linux-based systems, which are widely used in various IT environments, including cloud computing, web servers, and supercomputers. This knowledge is highly sought after by employers, making it a beneficial skill for career advancement in Information Technology.

II COMPETENCY

Apply foundational Linux knowledge, command-line skills, and system configuration techniques to administer and troubleshoot Linux environments effectively. Graduates will be able to manage users and groups, handle file systems, configure networks, and automate tasks using scripting. They will also gain experience with package management, system monitoring, and security practices within a Linux environment.

III COURSE OUTCOMES (COs)

After the completion of course based learning Students will be able to,

Course Outcome Number	Course Outcome Statement
1	Apply the concepts of open source technology and operating system fundamentals to install and use an open source operating system in practical scenarios.
2	Implement various Linux Command Line administration tasks and perform file,user, group,process management, storage and network management tasks.
3	Configure servers for front end and backend services and implement SHELL programming in order to devise a SHELL script to solve the problem.
4	Apply security measures to protect the operating environment and explain virtualization and their role in elastic computing

IV TEACHING-LEARNING & ASSESSMENT SCHEME

				Learning Scheme						Assessment Scheme													
				Actual Contact Hrs/Week			SL	Total	Credits		Theory					Practical				SLA		Total
Sr. No.	Course Title	Course Type	Course Code							Exam	CIA		ESE	Total		CIAP	ESEP	Total				
								Notional		Duration	MSE	CCE										
				CL	TL	LL		Hours		in Hrs	Max	Max	Max	Max	Min	Max	Max	Max	Min	Max	Min	
1	Skill-based Lab IV- Linux Server Administration Lab	-	ECLR1506	-	-	4	-	60	2	2	-	-	-	-	-	25	25	50	20	-	-	50

V LABORATORY LEARNING OUTCOME AND SUGGESTED LIST OF EXPERIMENTS (Minimum 8)

Sr. No.	Laboratory Learning Outcome	Laboratory Experiment Titles	Relevant COs
1	Installing Red Hat, or CentOS equips users with foundational skills in system administration, from installation to troubleshooting and security.	To Install the Red HAT/CentOs/Fedora Linux operating system	CO1
2	Learning and executing Linux commands for file and directory management helps users efficiently manage file system.It builds foundational skills in tasks like creating, deleting, moving, copying, and modifying files and directories using the command line.	To Learn and execute Linux commands for File and directory management.	CO1
3	Understand and apply Linux commands to effectively manage users, groups, and file permissions.	To Learn and execute Linux commands for managing Users, Groups, and Permissions.	CO2
4	Learn to identify, monitor, and control processes using essential Linux process management commands.	To Learn and execute Linux commands for Process management.	CO2
5	Understand and apply Linux commands to manage storage devices, partitions, and file systems. Gain hands-on experience in mounting, unmounting.	To Learn and execute Linux commands for managing Storage drives in the Linux environment.	CO2
6	Learn the fundamentals of shell scripting to automate tasks in a Linux environment	To write program using shell scripting	CO3
7	Understand and use Linux networking commands to configure and troubleshoot network settings.	To Learn and execute Linux commands for managing networking in Linux environment.	CO4
8	Learn to install and configure a C compiler in the Linux environment.Write, compile, and execute a basic "Hello World" program to understand the C programming workflow.	Installing C Compiler and executing a hello world program. (Content Beyond Syllabus)	CO4

VI LABORATORY ALIGNED COURSE CONTENT

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
1	CO1	1. Understand the need, advantages, and applications of Open Source, including key movements and development models. 2. Compare open-source and proprietary software, and identify major open-source licenses and their implications. 3. Gain practical knowledge of Linux installation and understand Linux system architecture. 4. Analyze the Linux boot process, file hierarchy (FHS), run-levels, kernel/shell concepts, and various shutdown methods.	**Introduction to Open-Source Software** 1.1 Need of Open Sources, Advantages and applications of Open sources, FOSS –FOSS usage, Free Software Movement, Open-Source Software Development Model, comparison with close source / Proprietary software, widely used open-source software license: Apache License 2.0, BSD license, GNU General Public License, MIT License, Mozilla Public License 2.0. 1.2 Installation of Linux (Redhat-CentOS-Fedora-Ubuntu): Linux Architecture, Kernel and shells, Boot Process, bootloader, understanding FHS of Linux, Understanding the different types of run-levels, understanding different types of shutdown commands.	10

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
2	CO2	1. Develop proficiency in using the Bash shell, including history, key commands, piping, redirection, and file system navigation. 2. Perform essential job and process management tasks, including system monitoring, scheduling jobs with CRON,and creating backups 3. Manage users and groups effectively, apply file and directory permissions, and utilize ACLs and attributes for secure access control. 4. Understand disk partitioning, create and manage file systems and logical volumes, configure swap space, and work with encrypted volumes. 5. Configure and troubleshoot Linux networking, set up SSH (including key-based authentication), and enable remote access using SSH and VNC.	**Open-Source Operating System: System Administration Task, Storage and Network Management** 2.1 Working with the Bash Shell, Getting the Best of Bash, Useful Bash Key Sequences, Working with Bash History, Performing Basic File System Management Tasks, Working with Files and Directories, Piping and Redirection, Finding Files, Working with Links. 2.2 Performing Job Management Tasks, System and Process Monitoring and Management, Managing Process Niceness, Scheduling Jobs using CRON, Creating Backups. 2.3 Managing Users and Groups, Commands for User Management, Managing Passwords, Modifying and Deleting User Accounts, Configuration Files, Creating Groups, Managing Permissions, the Role of Ownership, Basic Permissions: Read, Write, and Execute, Advanced Permissions, Working with Access Control Lists, Setting Default Permissions with umask, Working with Attributes. 2.4 Understanding Partitions and Logical Volumes, Creating Partitions, File Systems Overview, Creating File Systems, Mounting and Unmounting, File systems, Mounting File Systems Automatically Through fstab, Working with Logical Volumes, Creating Logical Volumes, Resizing Logical Volumes, Creating Swap Space, Working with Encrypted Volumes. 2.5 Understanding Network Manager, Network Manager Configuration Files, Network Service Scripts, Networking from the Command Line, Troubleshooting Networking, Setting Up IPv4 and IPv6, Configuring SSH, Enabling the SSH Server, Using the SSH Client, Using PuTTY on Windows Machines, Configuring Key-Based SSH Authentication, Using Graphical Applications with SSH, Using SSH Port Forwarding, Configuring VNC Server Access.	20

(Continued on next page)

Module Number	Mapped Course Outcome	Major Learning Outcomes	Theory Content	No. of Hours
3	CO3	1. Understand the concepts, benefits, and limitations of NFS and configure NFSv4, automount, and persistent mounts. 2. Set up and manage Samba file sharing, including advanced authentication and accessing Samba shares. 3. Configure and manage FTP servers and clients, and use basic FTP commands for file transfers and server access. 4. Install and configure a LAMP stack, create basic websites, manage Apache settings, use virtual hosts, and secure the server with TLS.	**Open-Source Operating System: Server Administration Task and Bash Shell Scripting** 3.1 What is NFS? Advantages and Disadvantages of NFS, Configuring NFS4, Setting Up NFSv4, Mounting an NFS Share, Making NFS Mounts Persistent, Configuring Automount, Configuring Samba, Setting Up a Samba File Server, Samba Advanced Authentication Options, Accessing Samba Shares, Understanding the features and advantages of FTP server, Configuring FTP server and FTP clients, Understanding FTP Basic Commands 3.2 Configuring LAMP stack: Configuring the Apache Web Server, creating a Basic Website, Understanding the Apache Configuration Files, Apache Log Files, Working with Virtual Hosts, Securing the Web Server with TLS Certificates, Setting Up MySQL and PhpMyAdmin.	16
4	CO4	1. Understand the fundamentals of SELinux, including contexts, booleans, tools, and port labeling to secure Linux systems. 2. Diagnose and resolve SELinux policy violations effectively for improved system security. 3. Learn to configure and manage FirewallD, its components, zones, custom services, and rich rules. 4. Gain knowledge on virtualization concepts, types, benefits, and implementation using KVM and Xen hypervisors.	**Open-Source Operating System: Advanced Security and Virtualization.** 4.1 SELinux Overview, SELinux Tools, SELinux Contexts, SELinux Booleans, Use SELinux port labeling to allow services to use non-standard ports, Diagnose and address SELinux policy violations, Configure FirewallD, Understand Firewalld Components, Setting Default Firewalld Zone, Creating Own Services in Firewalld, Assigning Services to Firewalld Zones, Adding Rich Rules for Network Range. 4.2 Introduction to virtualization and its types, need of virtualization, Benefits of Virtualization, Virtualization Implementation, Kernel based Virtual Machines (KVM) and XE.	14

VII SUGGESTED SELF LEARNING ACTIVITIES

NA

VIII SUGGESTED WEIGHTAGE TO ASSESSMENT PURPOSE

CO. No.	Course Outcome	BL1	BL2	BL3	BL4	BL5	BL6	Total
1	Apply the concepts of open source technology and operating system fundamentals to install and use an open source operating system in practical scenarios.	-	40 %	60 %	-	-	-	100%
2	Implement various Linux Command Line administration tasks and perform file,user, group,process management, storage and network management tasks.	-	30 %	70%	-	-	-	100%
3	Configure servers for front end and backend services and implement SHELL programming in order to devise a SHELL script to solve the problem.	-	50 %	50 %	-	-	-	100%
4	Apply security measures to protect the operating environment and explain virtualization and their role in elastic computing	-	20 %	40%	40%	-	-	100%

IX SUGGESTED LEARNING MATERIALS / TEXTBOOKS / REFERENCE BOOKS

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Linux: The Complete Reference	6th Edition	Richard Petersen	Tata McGraw Hill	2017
2	Linux Command Line and Shell Scripting Bible	3rd Edition	Richard Blum	Wiley	2015
3	Red hat Linux Networking and System Administration	3rd Edition	Terry Collings and Kurt Wall	Wiley	2005

Reference Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Linux Administration: A Beginner's Guide	3rd Edition	Wale Soyinka	Tata McGraw Hill	2020
2	Red Hat Enterprise Linux 6 Administration, Real World Skills for Red Hat Administrators	8th Edition	Sander van Vugt	John Wiley and Sons	2013
3	Rhcsa Red Hat Enterprise Linux 8: Training and Exam Preparation Guide	1st Edition	Asghar Ghori	Endeavor Technologies	2020

Online References

Sr. No	Website Name
A	https://archive.nptel.ac.in/courses/117/106/117106113/, NPTEL MOOC course on LINUX programming and scripting;

Mahavir Education Trust's
SHAH & ANCHOR
KUTCHHI ENGINEERING COLLEGE
An Autonomous Institute Affiliated to University of Mumbai

Address
Mahavir Education Chowk,
W.T. Patil Marg, Chembur
Mumbai- 400 088

विद्याधनं सर्वधनेषु प्रधानम्॥

www.sakec.ac.in
info@sakec.ac.in
@sakectweets
www.facebook.com/shahanchor
09136400555