South Indian Education Society's

Graduate School of Technology

NAAC 'A+' , NBA Accredited (EXTC,CE,IT)

Autonomous Institute Affiliated to University of Mumbai

SIES Graduate School of Technology (Autonomous)

RISE WITH EDUCATION

Department of Computer Engineering

Curriculum Structure for Undergraduate Academic Program in Computer Engineering

Third Year Syllabus

Board of Studies
Department of Computer Engineering

Academic Council
SIES Graduate School of Technology

Effective from: AY 2026-27

PREAMBLE

Dear Students, Faculty, and Stakeholders,

Greetings and congratulations to all the stakeholders of SIES Graduate School of Technology for getting autonomous status to the college from the year 2024-25. University Grant Commission vide letter No. F. 2- 10/2023(AC-Policy) dated 10 July 2024 conferred the autonomous status affiliated to University of Mumbai for a period of 10 years from the academic year 2024-2025 to 2033-2034 as per clause 7.5 of the UGC (Conferment of Autonomous Status Upon Colleges and Measures for Maintenance of Standards in Autonomous Colleges) Regulations,2023. Autonomy will help design and implement curriculum which is more learner centric and robust to cater to the changing need of Society and Industry.

The Department of Computer Engineering SIES Graduate School of Technology, Nerul, Navi Mumbai, was established in 2002 and the current intake capacity is 180.One of the notable achievements is accreditation by the National Board of Accreditation (NBA) in 2011,2021,2024 which reflects our commitment in maintaining high standards of education and infrastructure.
We are delighted to announce the introduction of the newly designed curriculum for autonomy at SIES Graduate School of Technology, The newly designed curriculum for the Computer Engineering course at our autonomous institute is a reflection of our commitment to excellence, innovation, and inclusivity in higher education. Guided by the principles and objectives of the National Education Policy (NEP), this curriculum aims to create a robust framework that not only equips students with in-depth technical knowledge but also fosters holistic development, ethical values, and a lifelong passion for learning.

Curriculum emphases on flexibility, multidisciplinary education, and a learner-centric approach. It aims to develop well-rounded engineers with a strong foundation in core computer engineering skills, while integrating emerging technologies such as AI, machine learning, and cybersecurity. The curriculum also offers flexibility through electives, allowing students to tailor their education and foster interdisciplinary learning for a comprehensive skill set.

This curriculum is designed with strong industry collaboration to provide students with real-world experience through internships and projects, ensuring they are industry-ready upon graduation. We promote lifelong learning, ensuring that students remain curious, adaptable, and committed to growth throughout their careers. This curriculum empowers students to become innovative engineers, ethical leaders, and global citizens who will contribute to the advancement of technology and the betterment of society.

We would like to express our sincere gratitude to the faculty, alumni, students, industry experts, academicians, and stakeholders for their continuous support in strengthening our academics, making SIESGST one of the top engineering colleges in the nation and the premier choice for engineering aspirants.

Chairperson
Board of Studies
Computer Engineering
SIES Graduate School of Technology

HEAD
Department of Computer Engineering
S.I.E.S. Graduate School of Technology
Sri Chandrasekarendra Saraswathy Vidyapuram
Plot-1-C & E, Sector-V, Nerul, Navi Mumbai-400706

Chairperson
Academic Council
SIES Graduate School of Technology

I/c PRINCIPAL
SIES GRADUATE SCHOOL OF TECHNOLOGY (AUTONOMOUS)
Plot 1C/D/E Sri Chandrasekarendra Saraswathy Vidyapuram
Sector V Nerul Navi Mumbai-400706

SIES GRADUATE SCHOOL OF TECHNOLOGY (Autonomous) Nerul Navi Mumbai 400706

Semester-wise Credit Distribution Structure for Four Year UG Engineering

Program – Computer Engineering

Semester		I	II	III	IV	V	VI	VII	VIII	Total Credits
Basic Science Course (BSC)	BSC/ESC	7	6		--	--	--	--	--	13
Engineering Science Course (ESC)		9	10		--	--	--	--	--	19
Program Core Course (PCC)	Program Courses	--	--	17	11	11	10	04	--	53
Program Elective Course (PEC)		--	--	--	--	04	04	07	--	15
Multidisciplinary Minor (MDM)	Multidisciplinary Courses		-		03	04	04	04	--	15
Open Elective (OE) Other than a particular program		--	--	--	--	--	--	03	03	06
Vocational and Skill Enhancement Course (VSEC)	Skill Courses	01	01	02	02	--	02	--	--	08
Ability Enhancement Course (AEC - 01, AEC-02)	Humanities Social Science and Management (HSSM)		02	--	--	02	--	--	--	04
Entrepreneurship/Economics/ Management Courses		--		02	02	--	--	--	--	04
Indian Knowledge System (IKS)			02		--	--	--	--	--	02
Value Education Course (VEC)		--	--	--	02	--	--	--	--	02
Research Methodology (RM)	Experiential Learning Courses	--	--	--	--	--	--	--	03	03
Community Engagement Project. / Field Project (FP)		--	--	01	01	--	--	-	-	02
Project		--	--	--	--	01	02	02	01	06
Internship/On Job Training (OJT)		--	---		--	--	--		12	12
Co-curricular Courses (CC)	Liberal Learning Courses	04			--	--	--	--	-	04
Total Credits (Major)		**21**	**21**	**22**	**21**	**22**	**22**	**20**	**19**	**168**

CURRICULUM STRUCTURE

THIRD YEAR ENGINEERING

(COMPUTER ENGINEERING)

ACADEMIC YEAR 2026-27

Nomenclature of the courses in the curriculum	
Abbreviation	**Title**
BSC	Basic Science Courses
ESC	Engineering Science Courses
PCC	Program Core Courses
PEC	Program Elective Course
MDM	Multidisciplinary Minor
OE	Open Elective
VSEC	Vocational and Skill Enhancement Course
AEC	Ability Enhancement Course
IKS	Indian Knowledge System
VEC	Value Education Course
RM	Research Methodology
CEP/FP	Community Engagement Project/Field Project
OJT	Internship/On Job Training
CC	Cocurricular Courses
ISE	In Semester Examination
MSE	Mid Semester Examination
ESE	End Semester Examination
CIAP	Continuous Internal Assessment Practical
ESEP	End Semester Examination Practical

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Program Structure for First Year Engineering

W.E.F. A.Y. 2024-25
Semester I

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
FEC101	Applied Mathematics -I	BSC	3	--	--	3	--	--	3
FEC1021/ FEC1022	Applied Physics/ Applied Chemistry @	BSC	3	--	--	3	--	--	3
FEC103	Basic Electrical & Electronics Engineering	ESC	2	--	--	2	--	--	2
FEC104	C-Programming	ESC	2	--	--	2	--	--	2
FEC105	Applied Mechanics and Robot Dynamics	ESC	2	--	--	2	--	--	2
FEL1011/ FEL1012	Applied Physics Lab/ Applied Chemistry Lab @	BSC	--	1	--	--	0.5	--	0.5
FEL102	Basic Electrical & Electronics Engineering Lab	ESC	--	2	--	--	1	--	1
FEL103	C-Programming Lab	ESC	--	2	--	--	1	--	1
FEL104	Applied Mechanics and Robot Dynamics Lab	ESC	--	2	--	--	1	--	1
FEL105	Engineering Workshop-I	VSEC	--	2	--	--	1	--	1
FEL106	Health, Wellness and Mindfulness	CC	--	2#+2	--	--	2	--	2
FEL107	Induction Cum Universal Human Values	CC	--	5*	--	--	2.5	--	2.5
	Total		**12**	**18**	--	**12**	**9**	--	**21**

Examination Scheme-FY Semester-I

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
FEC101	Applied Mathematics -I	20	20	60	3	--	--	100
FEC1021/ FEC1022	Applied Physics/ Applied Chemistry @	20	20	60	3	--	--	100
FEC103	Basic Electrical & Electronics Engineering	15	15	45	2	--	--	75
FEC104	C-Programming	15	15	45	2	--	--	75
FEC105	Applied Mechanics and Robot Dynamics	15	15	45	2	--	--	75
FEL1011/ FEL1012	Applied Physics Lab/ Applied Chemistry Lab@	--	--	--	--	25	--	25
FEL102	Basic Electrical & Electronics Engineering Lab	--	--	--	--	25	25	50
FEL103	C-Programming Lab	--	--	--	--	25	25	50
FEL104	Applied Mechanics and Robot Dynamics Lab	--	--	--	--	25	25	50
FEL105	Engineering Workshop-I	--	--	--	--	25	--	25
FEL106	Health, Wellness and Mindfulness	--	--	--	--	25	--	25
FEL107	Induction Cum Universal Human Values	--	--	--	--	25	--	25
Total		**85**	**85**	**255**		**175**	**75**	**675**

@Physics/Chemistry in one semester.
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.
Theory 1 credit for 1 hour and practical 1 credit for 2 hours.
*Indicates workload of a learner for UHV. Faculty Load: ½ hour per week per four groups
Two hours of practical class to be conducted for full class as demo/ discussion.
ISE: In Semester Examination**:** To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test etc. of 20/15 marks.
MSE: Mid Semester Examination: To be conducted as written examination for 20/15 marks of duration 1 Hr.
ESE: End Semester Examination
CIAP: Continuous Internal Assessment Practical. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).
ESEP: End Semester Examination Practical. Oral/Practical Examination will be conducted as End Semester Examination Practical (ESEP).

Program Structure for First Year Engineering
W.E.F. A.Y. 2024-25

Semester II

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
FEC201	Applied Mathematics -II	BSC	3	--	--	3	--	--	3
FEC2021/ FEC2022	Applied Physics/ Applied Chemistry @	BSC	3	--	--	3	--	--	3
FEC203	Engineering Graphics	ESC	2	--	--	2	--	--	2
FEC204	Digital System Design	ESC	3	--	--	3	--	--	3
FEC205	Professional Communication Techniques	AEC	2	--	--	2	--	--	2
FEL2011/ FEL2012	Applied Physics Lab/ Applied Chemistry Lab @	BSC	--	1	--	--	0.5	--	0.5
FEL202	Engineering Graphics Lab	ESC	--	2	--	--	1	--	1
FEL203	Digital System Design Lab	ESC	--	2	--	--	1	--	1
FEL204	Professional Communication Techniques Lab	AEC	--	1	--	--	0.5	--	0.5
FEL205	Object Oriented Programming Methodology Lab	ESC	--	2*+2	--	--	2	--	2
FEL206	Engineering Workshop-II	VSEC	--	2	--	--	1	--	1
FEL207	Indian Knowledge System	HSSM	--	2*+2	--	--	2	--	2
	Total		**13**	**16**	--	**13**	**8**	--	**21**

Examination Scheme-FY Semester-II

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
FEC201	Applied Mathematics -II	20	20	60	03	--	--	100
FEC2021/ FEC2022	Applied Physics/ Applied Chemistry @	20	20	60	03	--	--	100
FEC203	Engineering Graphics	15	15	45	03	--	--	75
FEC204	Digital System Design	20	20	60	03	--	--	100
FEC205	Professional Communication Techniques	15	15	45	02	--	--	75
FEL2011/ FEL2012	Applied Physics Lab/ Applied Chemistry Lab @	--	--	--	--	25	--	25
FEL202	Engineering Graphics Lab	--	--	--	--	25	25	50
FEL203	Digital System Design Lab	--	--	--	--	25	25	50
FEL204	Professional Communication Techniques Lab	--	--	--	--	25	--	25
FEL205	Object Oriented Programming Methodology Lab	--	--	--	--	25	25	50
FEL206	Engineering Workshop-II	--	--	--	--	25	--	25
FEL207	Indian Knowledge System	--	--	--	--	25	--	25
Total		**90**	**90**	**270**	**--**	**175**	**75**	**700**

@Physics/Chemistry in one semester.
* Two hours of practical class to be conducted for full class as demo/ discussion.
# Course evaluation is an activity based which may be an individual or group of students.
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.
Theory 1 credit for 1 hour and practical 1 credit for 2 hours.
ISE: In Semester Examination: To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test etc. of 20/15 marks.
MSE: Mid Semester Examination: To be conducted as written examination for 20/15 marks of duration 1 Hr.
ESE: End Semester Examination
CIAP: Continuous Internal Assessment Practical. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).
ESEP: End Semester Examination Practical. Oral/Practical Examination will be conducted as End Semester Examination Practical (ESEP).

Program Structure for Second Year

W.E.F. A.Y. 2025-26

Semester III

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
CEC301	Applied Mathematics III	PCC	3		--	3			3
CEC302	Data structure	PCC	3		--	3			3
CEC303	Discrete structure and Graph Theory	PCC	3		--	3			3
CEC304	Database Management system	PCC	3		--	3			3
CEC305	Computer Organization and Architecture	PCC	2		--	2			2
CEC306	Engineering Economics	HSSM	2		--	2			2
CEL301	Data structure Lab	PCC	--	2	--	--	1	--	1
CEL302	Database Management system Lab	PCC	--	2	--	--	1	--	1
CEL303	Computer Organization and Architecture Lab	PCC		2	--		1	--	1
CEL304	Skill Lab - Python Programming	VSEC	--	2*+2	--	--	2	--	2
CEM301	Mini Project 1A	CEP	--	2#	--	--	1	--	1
Total			**16**	**12**		**16**	**6**	**--**	**22**

Examination Scheme - CE Semester-III

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
CEC301	Applied Mathematics III	20	20	60	3	--	--	100
CEC302	Data structure	20	20	60	3	--	--	100
CEC303	Discrete structure and Graph Theory	20	20	60	3	--	--	100
CEC304	Database Management system	20	20	60	3	--	--	100
CEC305	Computer Organization and Architecture	15	15	45	2	--	--	75
CEC306	Engineering Economics	50	--	--	--	--	--	50
CEL301	Data structure Lab	--	--	--	--	25	25	50
CEL302	Database Management system Lab	--	--	--	--	25	25	50
CEL303	Computer Organization and Architecture Lab	--	--	--	--	25		25
CEL304	Skill Lab - Python Programming	--	--	--	--	25	25	50
CEM301	Mini Project 1A					25	25	50
Total		**145**	**95**	**285**	**--**	**125**	**100**	**750**

* Two hours of practical class to be conducted for full class as demo/ discussion.

# Indicates workload of a learner (Not faculty) for Mini Project 1A. Faculty Load: ½ hour per week per four groups

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours is of 60 marks and scaled to 45.

Program Structure for Second Year

W.E.F.A.Y.2025-26

Semester IV

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
CEC401	Applied Mathematics IV	PCC	3	--	--	3	--	--	3
CEC402	Operating System	PCC	3		--	3	--	--	3
CEC403	Analysis of Algorithm	PCC	3	--	--	3	--	--	3
CEC404	Critical Thinking and Design	HSSM	2	--	--	2	--	--	2
MDMC40X1	Multidisciplinary Minor (MDM -I)	MDM	3	--	--	3	--	--	3
CEL401	Operating System Lab	PCC	--	2	--	--	1	--	1
CEL402	Analysis of Algorithm Lab	PCC	--	2	--	--	1	--	1
CEL403	Skill Lab -Web Technology	VSEC	--	2*+2	--	--	2	--	2
CEL404	Value Education Course (UHV)	HSSM (VEC)	--	4	--	--	2	--	2
CEM401	Mini Project 1B	CEP	--	2#	--	--	1	--	1
Total			**14**	**16**		**14**	**7**	--	**21**

Examination Scheme - CE Semester-IV

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
CEC401	Applied Mathematics IV	20	20	60	3	--	--	100
CEC402	Operating System	20	20	60	3	--	--	100
CEC403	Analysis of Algorithm	20	20	60	3	--	--	100
CEC404	Critical Thinking and Design	15	15	45	2	--	--	75
MDMC40X1	Multidisciplinary Minor (MDM -I)	20	20	60	3	--	--	100
CEL401	Operating System Lab	--	--	--	--	25	25	50
CEL402	Analysis of Algorithm Lab	--	--	--	--	25	25	50
CEL403	Skill Lab -Web Technology	--	--	--	--	25	25	50
CEL404	Value Education Course (UHV)					50		50
CEM401	Mini Project 1B	--	--	--	--	25	25	50
Total		95	95	**285**	--	**150**	**100**	**725**

* Two hours of practical class to be conducted for full class as demo/ discussion.
UHV: Universal Human Values

# Indicates workload of a learner (Not faculty) for Mini Project 1B. Faculty Load: ½ hour per week per four groups.

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

Multidisciplinary Minor (MDM)

Data Science	Embedded Systems	Management
MDMC4021: Statistical Foundation for Data Science	**MDMC4031:** Microprocessor & Microcontroller	**MDMC4061:** Cost Management

Program Structure for Third Year
W.E.F.A.Y.2026-27

Semester V

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
CEC501	Theoretical Computer Science	PCC	3	--	--	3	--	--	3
CEC502	Software Engineering	PCC	3	--	--	3	--	--	3
CEC503	Computer Network	PCC	3	--	--	3	--	--	3
MDMC50X2	Multidisciplinary Minor (MDM-II)	MDM	3	--	--	3	--	--	3
CEPEC501X	Program Elective-I	PEC	3	--	--	3	--	--	3
CEL501	DevOps Lab	PCC	--	2	--	--	1	--	1
CEL502	Computer Network Lab	PCC	--	2	--	--	1	--	1
CEL503	Interpersonal and Career Skills	HSSM (AEC)	--	2*+2	--	--	2	--	2
MDML50X1	Multidisciplinary Minor (MDM-II) Lab/Tut[&]	MDM	--	2[&]	--	--	1[&]	--	1
CEPEL501X	Program Elective-I Lab	PEC		2			1		1
CEM501	Mini Project 2	Project	--	2[#]	--	--	1	--	1
Total			**15**	**14**		**15**	**7**	**--**	**22**

Examination Scheme - CE Semester-V

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE[$]	Exam Duration (Hrs.)			
		ISE	MSE					
CEC501	Theoretical Computer Science	20	20	60	3	--	--	100
CEC502	Software Engineering	20	20	60	3	--	--	100
CEC503	Computer Network	20	20	60	3	--	--	100
MDMC50X2	Multidisciplinary Minor (MDM-II)	20	20	60	3	--	--	100
CEPEC501X	Program Elective-I	20	20	60	3	--	--	100
CEL501	DevOps Lab	--	--	--		25	25	50
CEL502	Computer Network Lab	--	--	--		25	25	50
CEL503	Interpersonal and Career Skills	--	--	--		50		50
MDML50X1	Multidisciplinary Minor (MDM-II) Lab/ Tut[&]	--	--	--		25		25
CEPEL501X	Program Elective-I Lab					25		25
CEM501	Mini Project 2	--	--	--	--	25	25	50
Total		**100**	**100**	**300**		**175**	**75**	**750**

* Two hours of practical class to be conducted for full class as demo/ discussion.

# Indicates workload of a learner (Not faculty) for Mini Project 2. Faculty Load: ½ hour per week per four groups

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.

& _Indicates Tutorial only for the management track (MDMC5062)

Program Elective – I

Technology Bucket			
General	**Smart Systems**	**Network and Security**	**Artificial Intelligence**
CEPEC5011: Advanced Database Management System	**CEPEC5012:** Internet of things	**CEPEC5013:** Ethical Hacking	**CEPEC5014:** Data warehouse and Mining

Multidisciplinary Minor (MDM)

Data Science	Embedded Systems	Management
MDMC5022: Data Analytics & Visualization	**MDMC5032:** Embedded Systems and RTOs	**MDMC5062:** Supply Chain Management
MDML5021: Data Analytics & Visualization Lab	**MDML5031:** Embedded Systems and RTOs Lab	

Program Structure for Third Year
W.E.F.A.Y.2026-27

Semester VI

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
CEC601	System Programming and Compiler Construction	PCC	3	--	--	3	--	--	3
CEC602	Cryptography and Network Security	PCC	3	--	--	3	--	--	3
CEC603	Artificial intelligence and Soft Computing	PCC	3	--	--	3	--	--	3
MDMC60X3	Multidisciplinary Minor (MDM-III)	MDM	3	--	--	3	--	--	3
CEPEC601X	Program Elective-II	PEC	3	--	--	3	--	--	3
CEL601	Computational Intelligence Lab	PCC	--	2	--	--	1	--	1
CEL602	Skill Lab – Cloud Computing	VSEC	--	2*+ 2	--	--	2	--	2
MDML60X2	Multidisciplinary Minor (MDM-III) Lab/ Tut[&]	MDM	--	2[&]	--	--	1[&]	--	1
CEPEL601X	Program Elective-II Lab	PEC		2	--		1	--	1
CEP601	Major Project I	MJP	--	4[#]	--	--	2	--	2
Total			**15**	**14**		**15**	**7**		**22**

Examination Scheme - CE Semester-VI

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE[$]	Exam Duration (Hrs.)			
		ISE	MSE					
CEC601	System Programming and Compiler Construction	20	20	60	3	--	--	100
CEC602	Cryptography and Network Security	20	20	60	3	--	--	100
CEC603	Artificial intelligence and Soft Computing	20	20	60	3	--	--	100
MDMC60X3	Multidisciplinary Minor (MDM-III)	20	20	60	3	--	--	100
CEPEC601X	Program Elective-II	20	20	60	3	--	--	100
CEL601	Computational Intelligence Lab	--	--	--		25	25	50
CEL602	Skill Lab – Cloud Computing	--	--	--		25	25	50
MDML60X2	Multidisciplinary Minor (MDM-III) Lab/ Tut[&]	--	--	--		25		25
CEPEL601X	Program Elective-II Lab	--	--	--		25		25
CEP601	Major Project I					25	25	50
Total		**100**	**100**	**300**		**125**	**75**	**700**

* Two hours of practical class to be conducted for full class as demo/ discussion.
#Indicates workload of Learner (Not faculty), for Major Project I. Project Guide Load = ½ hour per week per project group
ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours are of 60 marks and scaled to 45.
& Indicates Tutorial only for the management track (MDMC6063)

Program Elective – II

Technology Bucket			
General	**Smart Systems**	**Network and Security**	**Artificial Intelligence**
CEPEC6011: Machine Vision	**CEPEC6012:** Robotics and Applications	**CEPEC6013:** Digital Forensics	**CEPEC6014:** Natural Language Processing

Multidisciplinary Minor (MDM)

Data Science	Embedded Systems	Management
MDMC6023: Decision Making and Business Intelligence	**MDMC6033:** Sensor Technology	**MDMC6063:** Organization Behaviour and Human Resource Management
MDML6022: Decision Making and Business Intelligence Lab	**MDML6032:** Sensor Technology Lab	

Program Structure for Fourth Year
W.E.F.A.Y.2027-28

Semester VII

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut	Theory	Pract.	Tut	Total
CEC701	Machine Learning	PCC	3	--	--	3	--	--	3
MDMC70X4	Multidisciplinary Minor (MDM-IV)	MDM	3	--	--	3	--	--	3
CEPEC701X	Program Elective – III	PEC	3	--	--	3	--	--	3
CEPEC702X	Program Elective-IV	PEC	3	--	--	3	--	--	3
OEC701X	Open Elective -I	OE	3	--	--	3	--	--	3
CEL701	Machine Learning Lab	PCC	--	2	--	--	1	--	1
MDML70X3	Multidisciplinary Minor (MDM-IV) Lab/ Tut[&]	MDM	--	2[&]	--	--	1[&]	--	1
CEPEL701X	Program Elective-III Lab	PEC	--	2	--	--	1	--	1
CEP701	Major Project Stage II	MJP	--	4[#]	--	--	2	--	2
Total			**15**	**10**		**15**	**5**		**20**

Examination Scheme - CE Semester-VII

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE[$]	Exam Duration (Hrs.)			
		ISE	MSE					
CEC701	Machine Learning	20	20	60	3	--	--	100
MDMC70X4	Multidisciplinary Minor (MDM-IV)	20	20	60	3	--	--	100
CEPEC701X	Program Elective – III	20	20	60	3	--	--	100
CEPEC702X	Program Elective-IV	20	20	60	3		--	100
OEC701X	Open Elective -I	20	20	60	3	--	--	100
CEL701	Machine Learning	--	--	--	--	25	25	50
MDML70X3	Multidisciplinary Minor (MDM-IV) Lab/ Tut[&]	--	--	--	--	25		25
CEPEL701X	Program Elective-III Lab	--	--	--	--	25	25	50
CEP701	Major Project Stage II					25	25	50
Total		100	**100**	**300**	--	**100**	**75**	**675**

#Indicates workload of Learner (Not faculty), for Major Project. Project Guide Load = ½ hour per week per project group.
$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours is of 60 marks and scaled to 45.
& _Indicates Tutorial only for the management track (MDMC7064)

Program Elective-III

Technology Bucket			
General	**Smart Systems**	**Network and Security**	**Artificial Intelligence**
CEPEC7011: High Performance Computing	**CEPEC7012:** Fog and Edge Computing	**CEPEC7013:** Blockchain Technology	**CEPEC7014:** Deep Learning

Program Elective-IV

Technology Bucket			
Data Science	**Smart Systems**	**Network and Security**	**Artificial Intelligence**
CEPEC7021: Augmented Reality and Virtual Reality	**CEPEC7022:** Quantum Computing	**CEPEC7023:** Intelligent Forensic	**CEPEC7024:** Reinforcement Learning

Multidisciplinary Minor (MDM)

Data Science	Embedded Systems	Management
MDMC7024: Big Data Analytics	**MDMC7034:** Industrial Internet of Things	**MDMC7064:** Marketing Management
MDML7023: Big Data Analytics Lab	**MDML7033:** Industrial Internet of Things Lab	

Open Elective-I

Course Code	Course Name
OEC7011	Project Management
OEC7012	Finance Management
OEC7013	Management Information System
OEC7014	Entrepreneurship Development and Management
OEC7015	Operation Research
OEC7016	Disaster Management and Mitigation Measures
OEC7017	Product Design

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Program Structure for Fourth Year
W.E.F.A.Y.2027-28

Semester VIII

Course Code	Course Name	Category	Teaching Scheme (Contact Hours)			Credits Assigned			
			Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
CEC801	Research Methodology	RM	3	-	--	3	-	--	3
OEC801X	Open Elective-II	OE	3	--	--	3	--	--	3
CEP801	Major Project Stage III	MJP	--	2#	--	--	1	--	1
CEINT801	Internship/Project/Research	Internship	--	--	--	--	12	--	12
Total			**6**	**2**		**6**	**13**		**19**

Examination Scheme - CE Semester-VIII

Course Code	Course Name	Examination Scheme						
		Theory				CIAP	ESEP	Total
		Internal Assessment		ESE$	Exam Duration (Hrs.)			
		ISE	MSE					
CEC801	Research Methodology	20	20	60	3		--	100
OEC801X	Open Elective-II	20	20	60	3		--	100
CEP801	Major Project Stage III					100	50	150
CEINT801	Internship/Project/Research					200		200
Total		40	**40**	**120**	--	**300**	**50**	**550**

# Indicates workload of a learner (Not faculty) for Major Project . Faculty Load: ½ hour per week per four groups

$ ESE of duration 3 hours are of 80 marks and scaled to 60. ESE duration of 2 hours is of 60 marks and scaled to 45.

Open Elective-II

Course Code	Course Name
OEC8011	Enterprise Resource Planning
OEC8012	Cyber Security and Laws
OEC8013	Energy Audit and Management
OEC8014	IPR and Patenting
OEC8015	Environmental Management
OEC8016	Digital Business Management
OEC8017	Human Resource Management

Multidisciplinary Minor (MDM)

Track	Minor Track	Partner Institute if any	Year/ Sem	Module	Code	Eligible
1	Machine Learning	SIES GST	SE/IV	Artificial Intelligence	MDMC4011	IT/EXTC/ CSE IOT
			TE/V	Machine Learning	MDMC5012	
				Machine Learning Lab	MDML5011	
			TE/VI	Natural Language Processing	MDMC6013	
				Natural Language Processing Lab	MDML6012	
			BE/VII	Deep Learning	MDMC7014	
				Deep Learning lab	MDML7013	
2	Data Science	SIES GST	SE/IV	Statistical Foundation for Data Science	MDMC4021	ECS/CE/EXTC
			TE/V	Data Analytics and Visualization	MDMC5022	
				Data Analytics and Visualization Lab	MDML5021	
			TE/VI	Decision Making and Business Intelligence	MDMC6023	
				Decision Making and Business Intelligence Lab	MDML6022	
			BE/VII	Big Data Analytics	MDMC7024	
				Big Data Analytics Lab	MDML7023	
3	Embedded Systems	SIES GST	SE/IV	Microprocessor and Microcontrollers	MDMC4031	CE/AIDS/AIML
			TE/V	Embedded systems and RTOS	MDMC5032	
				Embedded systems and RTOS Lab	MDML5031	
			TE/VI	Sensor Technology	MDMC6033	
				Sensor Technology Lab	MDML6032	
			BE/VII	Industrial Internet of Things	MDMC7034	
				Industrial Internet of Things Lab	MDML7033	
4	Cyber Security	SIES GST	SE/IV	Computer Network	MDMC4041	AIDS/AIML
			TE/V	Cryptography and Network Security	MDMC5042	
				Cryptography and Network Security Lab	MDML5041	
			TE/VI	Cloud Computing and Security	MDMC6043	
				Cloud Computing and Security Lab	MDML6042	
			BE/VII	Digital Forensics	MDMC7044	
				Digital Forensics Lab	MDML7043	
5	System Programming	SIES GST	SE/IV	Advance Data Structure	MDMC4051	CSEIOT/ECS/ IT
			TE/V	Advance Algorithm	MDMC5052	
				Advance Algorithm Lab	MDML5051	
			TE/VI	System Programming and Compiler Construction	MDMC6053	
				System Programming and Compiler Construction Lab	MDML6052	
			BE/VII	Distributed Systems	MDMC7054	
				Distributed Systems Lab	MDML7053	
6	Management	SIES SBS	SE/IV	Cost Management	MDMC4061	EXTC/CE/IT/ ECS/AIDS/ AIML/CSE IOT
			TE/V	Supply Chain Management	MDMC5062	
			TE/VI	Organizational Behaviour and Human Resource Management	MDMC6063	
			BE/VII	Marketing Management	MDMC7064	

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC501	Theoretical Computer Science	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEC501	Theoretical Computer Science	20	20	60	--	--	100

Pre- requisite:

1. CEC303: Discrete Structure and Graph theory

Program Outcomes Addressed

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of solutions
4. PO4: Investigations of Complex Problems
5. PO6: The Engineer and The World
6. PO7: Ethics
7. PO11: Life-Long Learning

Course Objectives: The course aims to enable students:

1. To acquire conceptual knowledge of grammar and languages.
2. To understand the relation between Regular Language and Finite Automata.
3. To understand the language hierarchy, CFG and CFL.
4. To design a PDA equivalent to a given context-free grammar/language.
5. To learn the principles of computation by designing a Turing Machine
6. To infer the knowledge of undecidable and NP class problems.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Use TCS theory to design regular expressions that represent regular languages.
2. Design, analyze, and optimize Finite Automata for language recognition.
3. Design Regular and Context Free Grammars and learn to simplify the CFG.
4. Design PDA for a given context-free grammar or language and enumerate its applications.
5. Design Turing machines as generators, deciders, and acceptors for various computational task
6. Utilize problem classification techniques for problem analysis.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Basics Concepts and Regular Languages**	**06**	
	1.1	Relations, Functions, Discrete Structures, Importance of TCS, Alphabets, Strings, Languages		**CO1**
	1.2	Regular operations, Regular Expression, Arden's theorem, RE Applications, Regular Language, Closure properties. Decision properties of RLs, Pumping lemma for RLs.		
		Self-learning Topics: RE in text search and replace, Application of Regular Languages in Compiler Design, Text Processing, and Natural Language Processing (NLP).		
2.0		**Finite Automata**	**08**	
	2.1	Finite Automata (FA) & Finite State machine (FSM).		**CO2**
	2.2	Deterministic Finite Automata (DFA) and Nondeterministic Finite Automata (NFA): Definitions, transition diagrams and Language recognizers, Equivalence between NFA with and ϵ- transitions, NFA to DFA Conversion, Minimization of DFA		
	2.3	FSM with output: Moore and Mealy machines, Applications and limitations of FA.		
		Self-learning Topics: State Elimination Method for converting FA to RE, Minimization of DFA using Equivalence Theorem, Conversion of Moore to Mealy & Mealy to Moore machine		
3.0		**Regular and Context Free Grammars**	**07**	
	3.1	Grammars and Chomsky Hierarchy Regular Grammar (RG), Equivalence of Left and Right linear grammar, Equivalence of RG and FA.		**CO3**
	3.2	Context Free Grammars (CFG): Definition, Sentential forms, Leftmost and Rightmost derivations, Parse tree, Ambiguity, Simplification of CFG: Eliminating unit productions, useless production, useless symbols, and Є-productions		
	3.3	Normal Forms: Chomsky Normal Form (CNF) and Greibach Normal Form (GNF), Context Free language (CFL) - Application: Parser, Markup languages; Pumping lemma, Closure properties.		
		Self-learning Topics: Left Recursion and Its Elimination, Applications of CFGs in XML Parsing, and Natural Language Processing (NLP).		
4.0		**Pushdown Automata (PDA)**	**07**	
	4.1	Definition, Language of PDA, PDA as generator, decider and acceptor of CFG, Deterministic PDA, Non- Deterministic PDA, Equivalence of PDA and CFG, Application of PDA.		**CO4**
		Self-learning Topics: Parsing & PDA: Top-Down Parsing, Bottom-up Parsing, Closure properties and Deterministic PDA.		
5.0		**Turing Machine (TM)**	**07**	

	5.1	Definition, Design of TM as generator, decider and acceptor, Variants of TM: Multitrack, Multitape, Universal TM, Applications, Power and Limitations of TMs.		**CO5**
		Self-learning Topics: Algorithms using Turing Machine, The Model of Linear Bounded Automata		
6.0		**Decidability and Computability**	**04**	
	6.1	Decidability and Undecidability, Recursive and Recursively Enumerable Language, Halting Problem, Rice's Theorem, Post Correspondence Problem.		**CO6**
		Self-learning Topics: NP Completeness of the SAT Problem, A Restricted Satisfiability Problem		
		Total	**39**	

Textbooks:

1. John E. Hopcroft, Rajeev Motwani, Jeffery D. Ullman, Introduction to Automata Theory Language and Computation, 3rd Edition, Pearson Education, 2008.
2. Michael Sipser, Theory of Computation, 3rd Edition, Cengage learning. 2013.
3. Vivek Kulkarni, Theory of Computation, Illustrated Edition, Oxford University Press, (12 April 2013) India.

Reference books:

1. J. C. Martin, Introduction to Languages and the Theory of Computation, 4th Edition, Tata McGraw Hill Publication, 2013.
2. Kavi Mahesh, Theory of Computation: A Problem-Solving Approach, Kindle Edition, Wiley-India, 2011.

Online References:

1. JFLAP: https://www.jflap.org/
2. Theory of Computation, IIT Kanpur: https://nptel.ac.in/courses/106104028
3. Theory of Computation, IIT Kanpur: https://nptel.ac.in/courses/106104148

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC502	Software Engineering	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEC502	Software Engineering	20	20	60	--	--	100

Pre- requisite:

1. FEC104: C Programming.

Program Outcomes Addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO5: Engineering tool Usage
5. PO6: The Engineer and the World
6. PO7: Ethics
7. PO8: Individual and Collaborative Team work
8. PO9: Communication
9. PO10: Project Management and Finance
10. PO11: Life-long Learning.

Course Objectives: The course aims to enable students:

1. To introduce fundamental concepts of Software Engineering, process models, and emerging trends in software development.
2. To develop the ability to analyze, elicit, document, and validate software requirements using structured techniques and modeling tools.
3. To provide knowledge of software estimation methods, metrics, project planning, scheduling, and management concepts.
4. To impart understanding of software design principles, UML diagrams, modular design, and architectural design approaches.
5. To build competency in software testing strategies, techniques, verification & validation, and software maintenance concepts.
6. To develop skills in software quality assurance, configuration management, and risk management processes in software projects.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Explain the nature of software, software engineering principles, various process models, agile approaches, and

advanced trends in software engineering.
2. Analyze and model software requirements using effective elicitation methods, scenario-based modeling, class-based modeling, DFDs, and SRS documentation standards.
3. Apply software metrics and estimation techniques, prepare project plans, schedules, and track the software projects.
4. Design software solutions using principles of cohesion, coupling, architectural styles, and UML diagrams
5. Implement appropriate testing strategies and maintenance techniques such as re-engineering and reverse engineering.
6. Identify risks, manage the changes to assure quality in software projects.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Software Engineering and Process Models**	**5**	
	1.1	Nature of Software and Software Engineering, Software Process Models and Capability Maturity Model (CMM), Advanced Trends in Software Engineering.		**CO1**
	1.2	Generic Process Model, Prescriptive Process Models: Waterfall, V-Model, Incremental, Evolutionary and Concurrent Models, Agile process and Agility Principles, Scrum.		
		Self-Learning Topics: Extreme Programming, Kanban		
2.0		**Software Requirements Engineering and Analysis**	**6**	
	2.1	Understanding the Requirements: Requirements Engineering, Eliciting Requirements, Negotiating Requirements, Requirements Monitoring, Validating Requirements.		**CO2**
	2.2	Requirement Modeling: Scenario Based Methods, class-based Methods, Data Flow Diagram, SRS (Software Requirement Specification) document format (IEEE).		
		Self-Learning topics: CRC Modelling		
3.0		**Software Estimation Metrics and Project Management**	**7**	
	3.1	Software Metrics, Software Project Estimation, Decomposition Techniques, Software Sizing, Problem-Based Estimation, LOC-Based Estimation, FP-Based Estimation, COCOMO II Model, Preparing Requirement Traceability Matrix.		**CO3**
	3.2	Project Management Concepts: Management Spectrum, people, product, process.		
	3.3	Project Scheduling: Basic Concepts, project scheduling, defining a task set for the software project, Scheduling: - Time-line chart, tracking the schedule, earned value analysis.		
		Self-Learning Topics: PERT, Estimation for Agile Development.		

4.0		**Software Design & UML Modelling**	**7**	
	4.1	Design Principles & Concepts, The Design Model, Modularity- Cohesion, and coupling. Architectural Design: Patterns, Design Decisions, Application Architectures.		**CO4**
	4.2	Designing Class-based Components, UI Design: The Golden Rules of UI Design, Interface Design Steps & Analysis, Design Evaluation.		
	4.3	Introduction to UML Modeling: Use Case Modeling, Structural modeling: Class diagram, Dynamic modeling: Sequence and Communication Diagram, Activity Diagram, Component and Deployment diagrams.		
		Self-Learning Topics: SOLID Object-Oriented Design Principles		
5.0		**Software Testing**	**8**	
	5.1	Software testing Strategies: A strategic approach to software testing, Test strategies for Conventional software: Unit Testing, Integration testing, Validation Testing, System testing, Verification and Validation		**CO5**
	5.2	Testing Conventional Applications: White-box testing: Basis path, Control structure testing, black-box testing: Graph based, Equivalence, Boundary Value		
	5.3	Types of Software Maintenance, Re-Engineering, Reverse Engineering		
		Self-Learning Topics: Agile Testing, Four-step strategy for real-time software testing		
6.0		**Software Configuration Management, Quality Assurance and Maintenance**	**6**	
	6.1	Quality Concepts, Software Quality Assurance, SQA metrics.		**CO6**
	6.2	Software Configuration Management: -SCM Repository, SCM Process.		
	6.3	Risk Management: Reactive versus Proactive risk strategies, Risk Identification, Risk projection, and RMMM plan.		
		Self-Learning Topics: Review Techniques – FTR, Walkthrough		
		Total	**39**	

Textbooks:

1. Roger Pressman, “Software Engineering: A Practitioner‘s Approach”, 9th edition , McGraw-Hill Publications, 2019.
2. Ian Sommerville, “Software Engineering”, 9th edition, Pearson Education, 2011
3. Ali Behfrooz and Fredeick J. Hudson, "Software Engineering Fundamentals", Oxford University Press.
4. Grady Booch, James Rambaugh, Ivar Jacobson, “The unified modeling language user guide”, 2nd edition, Pearson Education.

Reference books:

1. Pankaj Jalote, "An integrated approach to Software Engineering", 3rd edition, Springer.
2. Rajib Mall, "Fundamentals of Software Engineering", 5th edition, Prentice Hall India, 2018.
3. Ugrasen Suman, “Software Engineering – Concepts and Practices”, Cengage Learning, 2nd edition, 2022.

Online References:

1. Nptel Course on Software Engineering: Software Engineering - Course
2. Nptel course on Introduction to Software Engineering Challenges: nptel.ac.in/courses/106101061

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC503	Computer Network	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEC503	Computer Network	20	20	60	--	--	100

Pre- requisite: : None

Program Outcomes Addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO4: Conduct investigation of complex problems
5. PO5: Engineering Tool Usage
6. PO6: The Engineer and the world

Course Objectives: The course aims to enable students:

1. To interpret the issues and challenges of protocols design while delving into TCP/IP protocol suite.
2. To introduce concepts and fundamentals of physical layer
3. To describe various functions of Data Link Layer (DLL).
4. To compare the strengths and weaknesses of various routing algorithms.
5. To discuss various transport layer protocols.
6. To discuss various application layer protocols.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Discuss the concepts of data communication at physical layer and Compare ISO – OSI model with TCP/IP model.
2. Describe the various functions of Physical Layer.
3. Illustrate different design issues and error detection and correction mechanisms at data link layer.
4. Construct networks using IP addressing and sub-netting / super-netting schemes.
5. Apply transport layer protocols and congestion control algorithms to network scenarios.
6. Interpret the protocols in the application layer.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Networking**	**6**	
	1.1	Definition, Types of Networks: Local area networks (LAN), Metropolitan area networks (MAN), Wide area networks (WAN), Wireless networks, Networks Software, Protocol, Design issues for the Network layers.		**CO1**
	1.2	Network Models: The OSI Reference Model, TCP/IP Model, Network Topologies, Types of Transmission Medium.		
	1.3	Network Architectures: Client-Server, Peer To Peer, Hybrid. Network Devices: Bridge, Switch, Router, Gateway, Access Point.		
		Self-Learning: Evolution of computer networks (1G to 5G to 6G), Introduction to SDN (Software Defined Networking)		
2.0		**Data Link Layer**	**8**	
	2.1	Introduction and functions. Design Issues: Services Network Layer, Framing. ARQ strategies: Error Detection and correction, Parity Bits, Hamming Code, and CRC. Flow Control Protocols: Unrestricted Simplex, Stop and Wait, Sliding Window Protocol.		**CO2**
	2.2	MAC Sub layer: Multiple Access Protocols: Pure and Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA, Introduction to Ethernet IEEE 802.3, IEEE 802.11 a/b/g/n, IEEE 802.15 and IEEE 802.16 Standards.		
		Self-Learning: Link Layer Security, Wi-Fi security mechanisms: WEP, WPA, WPA2, WPA3		
3.0		**Network Layer**	**8**	
	3.1	Introduction: Functions of Network Layer, Switching Techniques: Circuit Switching, Message Switching, Packet Switching.		**CO3**
	3.2	IP Protocol: Classes of IP (Network addressing), IPv4, IPv6, Network Address Translation, Sub-netting, CIDR.		
	3.3	Network layer Protocols: ARP, RARP, ICMP, IGMP.		
		Self-Learning: IPv6 requirements for IoT, Case Study for the Institute network Design.		
4.0		**Network layer Routing Protocols**	**5**	
	4.1	Network Routing and Algorithms: Static Routing, Dynamic Routing, Distance Vector Routing, Link State Routing, Path Vector. Routing Protocols: RIP, OSPF, BGP. Routing in MANET: AODV, DSR.		**CO4**
		Self-Learning: EIGRP (Enhanced Interior Gateway Routing Protocol), Routing in data-centre networks (Clos, Fat-Tree topologies), SDN- based routing concepts.		

5.0		**Transport Layer**	**6**	
	5.1	Process to Process Delivery, Services, and Socket Programming. Elements of Transport Layer Protocols: Addressing, Connection establishment, Connection release, Flow control and buffering, Multiplexing, Congestion Control. Transport Layer Protocols: TCP and UDP, SCTP, RTP, Congestion control and Quality of Service (QoS), Differentiated services, TCP and UDP for Wireless networks.		**CO5**
		Self-Learning: TCP Variants: Tahoe, Reno, Cubic, BBR, QUIC protocol (by Google), Reliable data transfer in wireless networks		
6.0		**Application Layer**	**6**	
	6.1	Introduction, HTTP, HTTPs Web Caching, DNS, Email: SMTP, MIME, POP3, Webmail, FTP, TELNET, DHCP, SNMP.		**CO6**
		Self-Learning: DNSSEC (Secure DNS), CDN concepts (Akamai, Cloudflare, edge caching)		
		Total	**39**	

Textbooks:

1. Fourauzan B.,"Data Communications and Networking", 6th Edition, Tata McGraw- Hill, Publications, ISBN:0–07 – 058408 – 7
2. Andrew S. Tanenbaum, Computer Networks, 5th Edition, Pearson India, 2012.
3. Kurose, Ross, “Computer Networking a Top-Down Approach Featuring the Internet”, Pearson, ISBN-10: 0132856204
4. William Stallings, Data and Computer Communications, 10th Edition, Pearson Education, 2013.

Reference books:

1. L. Peterson and B. Davie, “Computer Networks: A Systems Approach”, 5th Edition, Morgan-Kaufmann, 2012.
2. Douglas E. Comer & M.S Narayanan, "Computer Network & Internet”, Pearson Education.
3. Nader F. Mir, Computer and Communication Networks, Second Edition, Prentice Hall, 2014.

Online References:

1. NPTEL- Computer Networks And Internet Protocol, IIT Kharagpur: nptel.ac.in/courses/106105183
2. NPTEL-Advanced Computer Networks: Advanced Computer Networks - Course

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC5022	Data Analytics and Visualization	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
MDMC5022	Data Analytics and Visualization	20	20	60	--	--	100

Pre- requisite:

1. CEC301: Applied Mathematics III
2. CEC401: Applied Mathematics IV
3. CEL304: Skill Lab - Python Programming

Program Outcomes Addressed:

1. PO1: Engineering Knowledge:
2. PO2: Problem Analysis
3. PO4: Conduct Investigations of Complex Problems
4. PO5: Engineering Tool Usage
5. PO11: Lifelong Learning

Course Objectives: The course aims to enable students:

1. To Introduce the concept of Data Analytics Lifecycle.
2. To Develop data modeling using Regression and Time series Techniques.
3. To define geospatial data and geospatial information.
4. To create awareness about Text analytics and its applications.
5. To provide overview of Data analytics and visualization with R/ Python.
6. To understand data visualization using visualization tools.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Apply the data analytics lifecycle to prepare data and build analytical models.
2. Apply Regression and Timeseries models on a given data set and perform prediction.
3. Illustrate various types of sources for raster and vector data.
4. Analyze Text data and gain insights.
5. Experiment with different analytics techniques and visualization using R / Python
6. Develop interactive dashboards integrating data transformation and visualization for professional reporting.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Data analytics and Life cycle**	**04**	
	1.1	**Data Analytics Lifecycle Overview:** Key roles, project background, business understanding, hypothesis creation, and identification of data sources.		**CO1**
	1.2	**Data Preparation & Modeling:** Data conditioning, Extract–Transform–Load (ETLT), exploration, visualization, variable selection, and model planning/building using common analytical tools.		
	1.3	**Results & Deployment:**Communicating insights effectively and operationalizing the analytics solution for real-world implementation.		
		Self-learning Topics: Case studies of end-to-end analytics projects, CRISP-DM vs Data Analytics Lifecycle		
2.0		**Regression Models and Time Series**	**08**	
	2.1	**Regression:** Regression equation, **assessing** model fit and assumptions, Cross-validation and model selection, Stepwise regression methods, Prediction using regression models.		**CO2**
	2.2	**Logistic Regression and Comparison:** Logistic Response function and logit, Logistic regression and Generalized Linear model, ROC, AUC, and confusion matrix for model evaluation Linear and logistic regression: similarities and differences.		
	2.3	Overview of Time Series Analysis Box-Jenkins Methodology, ARIMA Model Autocorrelation Function (ACF), Autoregressive Models, Moving Average Models, ARMA and ARIMA Models, Building and Evaluating an ARIMA Model, Reasons to Choose and Cautions		
		Self-learning Topics: Introduction to GLM families beyond logistic (Poisson, Gamma), GARCH models for volatility forecasting (finance)		
3.0		**Geospatial Data Analysis**	**08**	
	3.1	Introduction to Geospatial Data, Sources of Raster Data, Sources of Vector Data, Overview of Geospatial Data Analysis Tools, Overview of Geospatial Programming languages.		**CO3**
		Self-learning Topics: Case studies of applications where raster and vector data is used.		
4.0		**Text Analytics**	**06**	
	4.1	History of text mining, Roots of text mining overview of seven practices of text analytic, Application and use cases for Text mining: extracting meaning from unstructured text, Summarizing Text.		**CO4**
	4.2	Text Analysis Steps, A Text Analysis Example, Collecting Raw Text, Representing Text, Term Frequency—Inverse Document Frequency		
		(TFIDF), Categorizing Documents by Topics, Determining Sentiments, Gaining Insights.		

		Self-learning Topics: Case studies on healthcare (clinical notes), Ecommerce (reviews, rating analysis)		
5.0		**Data Analytics and visualization with R / Python**	**06**	
	5.1	Introduction to R: Data Import and Export, Attribute and Data type, Descriptive statistics, Visualization and Analysis using R.		**CO5**
	5.2	Essential Data Libraries for data analytics and Visualization in Python, Visualization and Analysis using Python.		
		Self-learning Topics: Case studies on simple analytical reports using Python or R, Introduction to dataset import workflows in R/Python		
6.0		**Introduction to Data Visualization Tools**	**07**	
	6.1	Tool interfaces, data connections, data cleaning and modeling, building interactive dashboards with filters, slicers, and publishing reports. Principles of effective data visualization, chart selection, and visual design.		**CO6**
	6.2	Basics of DAX, calculated columns, measures, and essential time-intelligence functions.		
	6.3	Storytelling with data for business insights and ethical visualization practices. Interactive reporting with filters, bookmarks, and performance optimization.		
		Self-learning Topics: Introduction to connecting different data sources in dashboards		
		Total	**39**	

Textbooks:

1. D. Dietrich, Data science & big data analytics : discovering, analyzing, visualizing and presenting data. Indianapolis, IN: Wiley, 2015.
2. B. Motwani, Data Analytics Using Python. New Delhi, India: Wiley Publications, 2019.
3. B. Motwani, Data Analytics Using R. New Delhi, India: Wiley Publication, 2015.
4. P. Bruce, A. Bruce, and P. Gedeck, Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python, 2nd ed., Sebastopol, CA, USA: O'Reilly Media, 2020.
5. G. Miner and T. Hill, Practical Text Mining and Statistical Analysis for Non-Structured Text Data Applications, 1st ed., Burlington, MA, USA: Academic Press, 2012.

Reference books:

1. J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed., Waltham, MA, USA: Morgan Kaufmann, 2011.
2. W. McKinney, Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter, 3rd ed., Sebastopol, CA, USA: O'Reilly Media, 2022.

Online References:

1. NPTEL- Introduction to Data Analytics: https://nptel.ac.in/courses/110106072
2. Coursera-Data Visualization with Python: Data Visualization with Python | Coursera

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC5032	Embedded Systems and RTOS	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
MDMC5032	Embedded systems and RTOS	20	20	60	--	--	100

Pre- requisite:

1. FEC204: Digital System Design
2. FEC104: C programming
3. MDMC4031: Microprocessors & Microcontrollers

Program Outcomes Addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO6: The Engineer and The World
5. PO7: Ethics
6. PO11: Life-Long Learning

Course Objectives: The course aims to enable students:

1. Interpret embedded system fundamentals, classifications, design metrics, and real-world applications.
2. Explore embedded processing cores, memory architectures, and ARM Cortex-M3 processor features.
3. Interface sensors and actuators using standard communication protocols and implement low-power, battery-aware embedded designs.
4. Comprehend safety and security standards including IEC 61508, IEC 60601, ISO 26262, and ISO 21434 for cybersecurity in embedded systems.
5. Apply RTOS principles, including task management, inter-process communication, memory management, and unit testing in embedded applications.
6. Implement real-time scheduling algorithms (RMS, EDF) and develop RTOS-based embedded applications.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Explain embedded system fundamentals, classifications, design metrics, and applications in real-world scenarios.
2. Describe embedded processing cores, memory types, and ARM Cortex-M3 architecture for system design.
3. Interface sensors and actuators using communication protocols (UART, SPI, I2C) and apply low-power and battery-aware design techniques.
4. Analyze and compare IEC 61508, IEC 60601, ISO 26262 and evaluate awareness of cybersecurity needs in

embedded systems.

5. Apply task management, inter-process communication, and memory management mechanisms in an RTOS environment.
6. Apply real-time scheduling algorithms to analyze system schedulability and develop practical RTOS-based embedded applications.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Embedded Systems Fundamental**	**04**	
	1.1	Definition, Characteristics, Classification, Applications		**CO1**
	1.2	Design metrics of Embedded system (performance, power, cost, size)		
	1.3	Challenges and design trade-offs		
		Self-learning Topics: 8-bit vs 32-bit MCU comparison		
2.0		**Embedded Systems Hardware & Programming**	**09**	
	2.1	Embedded system architecture overview (MCU, SoC, SoC components)		**CO2**
	2.2	Basics of ARM Cortex-M3 processor		
	2.3	Memory types in embedded systems (Flash, SRAM, EEPROM, Cache)		
	2.4	Embedded C Concepts: Bitwise operations for port control, GPIO programming fundamentals, Polling vs Interrupt handling in C, Writing delay routines, LED blinking, switch interfacing		
		Self-learning Topics. Datasheet exploration exercise		
3.0		**Communication & Low Power Systems**	**07**	
	3.1	UART, I2C, SPI basics		**CO3**
	3.2	Wireless Interface overview: BLE / Wi-Fi		
	3.3	Sensor and Actuator basics with examples (temp, motor)		
	3.4	Low-power concepts (Sleep modes, duty cycling)		
	3.5	Battery and Power Management: Common battery types used in embedded systems, Power-saving modes (sleep/idle modes) in embedded systems		
		Self-learning Topics: ADC programming example		
4.0		**Industry Standards and Secure Coding**	**05**	
	4.1	Overview of Standards: IEC 61508 (Functional Safety), IEC 60601 (Medical Electrical Equipment), ISO 26262 (Automotive Safety)		**CO4**
	4.2	Basic lifecycle steps (Requirement → Development → Testing→ Verification & Validation)		
	4.3	Fundamentals of secure coding, common vulnerabilities, and unit testing		
		Self-learning Topics: Safety Critical Software Engineering under		

		Functional Safety Standards		
5.0		**RTOS Concepts & Programming**	**09**	
	5.1	Concepts: Task, Thread, Multitasking, Latency		**CO5**
	5.2	Inter-task communication: Semaphore, Mutex, Queue		
	5.3	Task scheduling basics: Static vs Dynamic scheduling		
	5.4	FreeRTOS programming fundamentals: Task creation, Delay & periodic tasks, Queue send/receive, Semaphore to avoid race conditions		
		Self-learning Topics: RTOS-Based Application Design: Smart Home, Automatic Fan Controller, Motor speed control		
6.0		**Real Time Scheduling and Analysis**	**05**	
	6.1	Scheduling: Rate Monotonic Scheduling (RMS) & Earliest Deadline First Scheduling (EDF)		**CO6**
	6.2	Schedulability formula + simple numerical problems		
		Self-learning Topics: Additional real-time scheduling algorithms		
		Total	**39**	

Textbooks:

1. K. V. K. Prasad, Embedded or Real-Time Systems: Concepts, Design and Programming Black Book. New Delhi, India: Dreamtech Press, 2003.
2. R. Mall, Real-Time Systems: Theory and Practice. New Delhi, India: Prentice Hall, 2009.
3. Rajkamal, Embedded System: Architecture, Programming and Design. New Delhi, India: Tata McGraw-Hill, 2011.
4. J. Yiu, The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, 3rd ed. Oxford, UK: Elsevier, 2014.

Reference books:

1. D. Simon, An Embedded Software Primer, New Delhi, India: Pearson, 2009.
2. J. W. Valvano, Embedded Microcomputer Systems: Real-Time Interfacing, 3rd ed. Boston, MA, USA: Cengage Learning, 2012.
3. Mastering FreeRTOS: Hands-On Real-Time Operating Systems. Birmingham, UK: Packt Publishing.
4. FreeRTOS Official Documentation (AWS FreeRTOS).
5. X. Fan, Real-Time Embedded Systems: Design Principles and Engineering Practices. Oxford, UK: Newnes (Elsevier).

Online References:

1. MOOC Course (NPTEL): https://onlinecourses.nptel.ac.in/noc20_cs16/preview
2. Real Time Operating System by Rajib Mall (IIT Kharagpur): https://onlinecourses.nptel.ac.in/noc25_cs78/preview
3. Embedded Systems (Real-Time Embedded Systems, IIT Kharagpur): https://nptel.ac.in/courses/108105057

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question 1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question 3 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC5062	Supply Chain Management	03	-----	01	03	------	01	04

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
MDMC5062	Supply Chain Management	20	20	60	25	------	125

Pre- requisite:

1. Basic MS Excel.

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO7: Global and Ethical Practices
4. PO8: Collaborative Teamwork
5. PO9: Communication
6. PO11: Lifelong Learning

Course Objectives: The course aims to enable students:

1. To introduce the fundamental concepts, components, and processes of supply chain management in manufacturing and service sectors.
2. To develop understanding of demand forecasting, inventory management, procurement, and logistics as core elements of SCM.
3. To equip students with the ability to analyze supply chain networks, evaluate performance, and identify improvement opportunities.
4. To familiarize students with supply chain technologies such as ERP systems, RFID, IoT, and analytics tools.
5. To build decision-making skills related to sourcing, production planning, transportation, and distribution strategies.
6. To promote awareness of sustainability, risk management, and global challenges in supply chain operations.

Course Outcomes: Upon completion of this course, Students will be able to:

1. To explain key supply chain, logistics, and international shipping concepts.
2. To apply analytical tools to solve basic warehousing, transportation, and network design problems.
3. To analyse supply chain flows and identify performance gaps, risks, and integration challenges.
4. To evaluate supply chain efficiency using cost, service, sustainability, and responsiveness metrics.
5. To design technology-enabled and resilient supply chain solutions for domestic and global environments.
6. To demonstrate understanding of sustainable, ethical, and technology-enabled supply chain practices and assess their impact on organizational performance.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Module 1: Introduction to Supply Chain, Logistics and International Shipping**	**09**	
	1.1	Introduction to Supply Chain, Decision Phases in a Supply Chain, Linkages and Decisions in SCM. Process and Cycle View of a Supply Chain. Supply Chain Models: Efficient Chain, Responsive Supply Chain and Agile Models. Importance of supply chain flows. Logistics and Shipping: functions. Objectives, Goals, Decisions. 1st Party, 2nd Party, 3rd Party, 4th Party Logistic service provider Introduction to International Shipping: Freight Forwarders, shipping documents, Ports and customs, Incoterms Reverse Logistics.		**CO1, CO2**
		Self-learning Topics: Importance of supply chain flows. Logistics and Shipping: functions. Objectives, Goals, Decisions		
2.0		**Warehousing and Transportation**	**09**	
	2.1	Warehouse: role, type, functions. Warehouse location modelling: methods such as COG, MCOG, and the LP approach. Layout design, Space calculation, Warehouse automation, Hub and Spoke Model, WMS Distribution, Role, Importance, Levels, Channels, Structure, Functions. Channel partners, functions. Infrastructure, road, rail, air, water, pipeline. Freight Management, Freight cost. Transportation Network Route Planning, Containerization, and Packing. Effective / Cost Optimising strategies.Cross-docking, Milk run, and transhipment.		**CO2, CO3**
		Self-learning Topics: Role, Importance, Levels, Channels, Structure, Functions. Channel partners, functions.		
3.0		**Performance Measurement and Controls in Supply Chain Management**	**03**	
	3.1	Supply Chain Cost Calculations, Supply Chain Reliability & Responsiveness Supply Chain Asset Management: cash to cash cycle, return on working capital & working capital, Analytics Framework Based on SCOR Model		**CO3, CO4**
		Self-learning Topics: Concepts of Supply Chain Asset Management: cash to cash cycle & working capital		
4.0		**Designing a network in a supply chain**	**06**	
	4.1	Network design in the supply chain, Alternate channels in distribution, Location decisions in the supply chain using Excel, Role and factors affecting network designing, Network Optimisation Model, Network design in an uncertain environment, Flexibility in Supply Chain, Practical problems using Excel		**CO2, CO4 CO5**
		Self-learning Topics: Role and factors affecting network designing		

5.0		**Supply Chain Integration and Designing Global Supply Chain Networks**	**06**	
	5.1	Bullwhip effect, Collaborative Planning Forecasting Replenishment (CPRF) concept. Inventory Management and Risk Pooling, Strategic Alliances, Retailer-Supplier Partnerships: Types, Requirements, and Inventory Ownership. Outsourcing and related decisions Global Market: Technological/ Cost/ Political, and Economic Forces. Risks and Advantages of the International Supply Chain. International versus Regional products. Local autonomy versus central control. Regional differences in Logistics- Cultural differences/ infrastructure/ performance expectation, and evaluation, Information systems availability, and human resources.		**CO3, CO5**
		Self-learning Topics: Concepts related to Outsourcing and related decisions, Risks and Advantages of the International Supply Chain.		
6.0		**Ethical issues in Supply Chain and Technological Trends in Supply Chain**	**06**	
	6.1	Supply chain vulnerability. Conformance to applicable laws such as Contract and commercial laws, Trade regulation, government procurement regulations, patents, Copyrights, trademark laws, transportation and logistics laws and regulations, Environmental laws. International practices. Confidentiality and proprietary information. Functional application of technologies in Supply Chain: Blockchain, AI, IOT, RFID, AR, VR, ERP Goldratt Supply Chains, Sustainable Supply Chain, Resilient supply chains, Green Supply chain, Lean supply chain.		**CO4, CO5**
		Self-learning Topics: Concepts of Blockchain, AI, IOT, RFID, AR, VR, ERP		
		Tutorial on application of supply chain management concepts in excel		
		Total	**39**	

Textbooks:

1. Supply Chain Management: Strategy, Planning, and Operation, 6th Edition, Sunil Chopra & Peter Meindl, Pearson Education

Reference books:

1. Supply Chain Analytics and Modelling: Quantitative Tools and Applications, 1st Edition, Kogan Page
2. Supply Chain Logistics Management, 2nd Edition, Bowersox, Donald J, 2014, McGraw Hills.
3. Supply Chain Management: The Basics & Beyond, Copacino, William C, 1997, ST. Lucie Press.
4. Logistics and Supply Chain Management, Martin Christopher

Online References:

1. The International Journal of Logistics Management https://www.emerald.com/ijlm
2. Supply Chain Management: An International Journal: https://www.emeraldgrouppublishing.com/journal/scm

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question 1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question 3 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC5011	Advanced Database Management System	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC5011	Advanced Database Management System	20	20	60	--	--	100

Pre- requisite:
1. CEC304 - Database Management System

Program Outcomes:
1. PO1: Engineering knowledge.
2. PO2: Problem analysis.
3. PO3: Design/development of solutions.
4. PO5: Engineering Tools usage.
5. PO11: Life-long learning.

Course Objectives: The course aims to enable students:
1. To provide insights into distributed database design.
2. To impart knowledge related to query processing and query optimization phases of a database management system.
3. To introduce the concepts of access control models (DAC, MAC, and RBAC).
4. To specify the various approaches used for using XML and JSON technologies.
5. To apply the concepts behind the various types of NoSQL databases and utilize them for MongoDB.
6. To learn about the trends in advanced databases.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Design distributed database architectures, apply data fragmentation, and concurrency control techniques.
2. Measure query costs and design alternate efficient paths for query execution.
3. Analyze and implement access control mechanisms.
4. Apply and demonstrate XML and JSON databases for efficient data representation and querying for better interoperability.
5. Compare different types of NoSQL databases.

6. Describe various trends in advanced databases.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Distributed Databases**	**08**	
	1.1	Introduction, Distributed DBMS Architecture, Data Fragmentation, Replication and Allocation Techniques for Distributed Database Design.		**CO1**
	1.2	Distributed Transaction Management: Definition, properties, types, architecture, Distributed Concurrency Control- Taxonomy, Locking based, Basic TO algorithm.		
		Self-learning Topics: Recovery in Distributed Databases: 2PC and 3PC protocol, Database indexing.		
2.0		**Query Processing and Optimization**	**07**	
	2.1	Query Processing: Overview, Layers/Phases of query processing in distributed databases, Measures of Query cost, Selection operation, Sorting, Join Operations, and other Operations, Evaluation of Expression,		**CO2**
	2.2	Query Optimization: Translations of SQL Queries into relational algebra, Heuristic approach & cost-based optimization		
		Self-learning Topics: Layers/Phases of query processing in a centralized database.		
3.0		**Advanced Database Access Protocols**	**06**	
	3.1	Introduction to Database Access Protocols, Authorization, Discretionary Access Control Based on Granting and Revoking Privileges, Mandatory Access Control, and Role-Based Access Control for Multilevel Security		**CO3**
		Self-learning Topics: Database Security, Flow Control.		
4.0		**Data interoperability – XML and JSON**	**06**	
	4.1	XML Databases: Document Type Definition, XML Schema, Querying and Transformation: XPath and XQuery.		**CO4**
	4.2	Basic JSON syntax (JavaScript Object Notation), JSON data types, Stringifying and parsing the JSON for sending & receiving, JSON Object retrieval using key-value pair, and jQuery, XML vs JSON		
		Self-learning Topics: MongoDB Shell.		
5.0		**NoSQL Distribution Model**	**08**	
	5.1	NoSQL database concepts: NoSQL data modeling, Benefits of NoSQL, comparison between SQL and NoSQL database systems.		**CO5**
	5.2	Replication and sharding, Distribution Models, Consistency in distributed data, CAP theorem, Notion of ACID Vs BASE, handling Transactions, consistency and eventual consistency		
	5.3	Types of NoSQL databases: Key-value data store, Document database, Column Family Data store and Graph Store, Comparison of NoSQL databases w.r.t. CAP theorem and ACID properties.		

		Self-learning Topics: Graph NoSQL data store tools.		
6.0		**Trends in advanced databases**	**4**	
	6.1	Temporal database: Concepts, time representation, time dimension, and incorporating time in relational databases.		**CO6**
	6.2	Spatial database: Introduction, data types, models, operators, and queries.		
		Self-learning Topics: Mobile databases, Multimedia databases.		
		Total	**39**	

Textbooks:

1. Korth, Siberchatz, Sudarshan, "Database System Concepts", 7th Edition, McGraw Hill ,2020–2022
2. Elmasri and Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson Education ,2017.
3. Ozsu, M. Tamer, Valduriez, Patrick, "Principles of distributed database systems", 3rd Edition, Pearson Education, Inc. ,2020.
4. Sadalage, Pramod J., and Martin Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. 1st Edition, Addison-Wesley Professional, 2012.
5. Jeff Friesen, Java, XML and JSON, Second Edition, 2019 Après, Inc.

Reference Books:

1. Chhanda Ray, Distributed Database System, Pearson Education, India.
2. Adam Fowler, NoSQL for Dummies, John Wiley & Sons, Inc.
3. Shashank Tiwari, Professional NOSQL, John Willy & Sons. Inc.

Online References:

1. NPTEL https://archive.nptel.ac.in/courses/106/104/106104135/
2. Scaler https://www.scaler.com/topics/dbms/distributed-database-in-dbms/
3. W3Schools https://www.w3schools.com/xml/default.asp
4. MongoDB https://www.mongodb.com/docs/manual/
5. PostGIS https://postgis.net/workshops/postgis-intro/

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question 1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question 3 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC5012	Internet of Things	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC5012	Internet of Things	20	20	60	--	--	100

Pre- requisite:

1. FEC104: C-Programming
2. CEL304: Skill Lab - Python Programming
3. CEC305: Computer Organization and Architecture

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design /Development of Solutions
4. PO4: Conduct Investigations
5. PO6: The Engineer and the world
6. PO11: Life Long Learning

Course Objectives: The course aims to enable students:

1. To introduce the fundamental concepts and architecture of the Internet of Things.
2. To introduce data handling methods and its tools for various IoT Applications.
3. To explore various Sensors and Actuators for any given Problem statement.
4. To introduce and understand IoT communication requirements across Physical, Datalink, Network and Application layers
5. To introduce the architecture of IoT systems, their unique constraints, and the importance of
6. security and privacy in IoT environments.
7. To explore real-world IoT applications in domains such as smart homes, agriculture, and e-health

Course Outcomes: Upon completion of this course, Students will be able to:

1. Summarize the fundamental concepts of Internet of Things.
2. Analyze various data handling methods in IoT-based systems using appropriate tools or Platforms.
3. Analyze physical, data link, and network layer protocols for IoT communication.
4. Analyze network layer and Application protocols for IoT communication
5. Explain IoT Security architecture, communication models and device characteristics. Also can identify fundamental security and privacy challenges.

6. Analyze real-world IoT applications in smart homes, agriculture, and e-health.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Internet of Things**	**06**	
	1.1	Introduction to IoT-Define IoT, Characteristics of IoT, Evolutionary phases of Internet, IoT and Digitization, IoT impact, Convergence of Information of Technology and Operational Technology, IoT challenges		**CO1**
	1.2	IoT World Forum (IoTWF) Standardized Architecture, Drivers behind new Network Architecture, Comparing IoT Architectures, Core IoT functional Stack and IoT Data Management and Compute stack.		
	1.3	Sensors, Actuators and Smart Objects -Definition, Characteristics and Trends, Wireless Sensor Network and its Design constraints.		
		Self-learning Topics: Micro-Electro-Mechanical Systems (MEMS)		
2.0		**IoT Data collection and Analytics**	**06**	
	2.1	Introduction to IoT data collection, types and sources of IoT Data (Environmental data, Location data, Equipment data…), IoT Data collection mechanisms (Packet-based, Flow based, Log based), IoT Data collection Architecture.		**CO2**
	2.2	IoT Data Acquisition & organization, Data Analytics in IoT, Trends and challenges in IoT Data Analytics.		
		Self-learning Topics: Data analytics tools such as Python with libraries like Pandas, NumPy, and Scikit-learn for data analysis and visualization. Learn about applying machine learning algorithms to IoT data for predictive analytics and anomaly detection. .		
3.0		**IoT Protocols and Standards-Physical and Data link layer**	**08**	
	3.1	Overview of IoT protocol stack, role of Standard Organization in IoT (ITU, IETF, IEEE and 3GPP), Comparison of IoT and Traditional Internet Protocols.		**CO3**
	3.2	Physical and Data link Layer protocols -IEEE 802.11(Wi-Fi), IEEE 802.15.4 (Zigbee,6LoWPAN), Lora WAN, NB-IoT.		
		Self-learning Topics: Case studies on protocol selection and deployment in smart energy, logistics, and healthcare IoT.		
4.0		**IoT Protocols and Standards-Network and Application Layer**	**08**	
	4.1	Network and Transport Layer protocols-IPV6 for IoT,6LoWPAN adaptation, Transport protocols-TCP, UDP in IoT		**CO4**
	4.2	Application Layer: MQTT, CoAP, DDS, HTTP & REST vs. SOAP		
		Self-learning Topics: Routing protocols: Flat, hierarchical, location-based approaches, Energy-efficient routing protocols (LEACH, PEGASIS, geographic routing)		

5.0		**IoT Security & Privacy**	**06**	
	5.1	Introduction to IoT security, privacy concerns in IoT, overview of security challenges, Security across IoT communication layers		**CO5**
	5.2	Privacy risks in IoT data collection, privacy-enhancing technologies (anonymization, pseudonymization)		
		Self-learning Topics: Develop a secure IoT healthcare monitoring system with encrypted communication and role-based access control.		
6.0		**IoT Applications**	**05**	
	6.1	IoT Applications Case studies are-Smart Home: Characteristics of Smart Home - Smart Home Energy Management, Smart Appliances, Communication Technologies for Smart Homes, maintenance, security, challenges.		**CO6**
	6.2	Smart Agricultural: characteristics and applications -Scarecrow, Smart Irrigation System, Crop Water Management, Integrated Pest Management, Sensor-based field and resource mapping, Remote equipment monitoring)		
	6.3	E-health: Characteristics of e-health and applications- monitoring of health parameters, smart medicine box, elderly people monitoring, challenges		
		Self-learning Topics: Build and deploy a machine learning-based predictive maintenance model using IoT sensor data on an edge device.		
		Total	**39**	

Textbooks:

1. Bahga and V. Madisetti, Designing the Internet of Things, 1st ed. Hyderabad, India: University Press, 2015.
2. R. Kamal, Internet of Things: Architecture and Design Principles, 1st ed. New Delhi, India: McGraw Hill Education, 2017.
3. D. Hanes and G. Salgueiro, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, 1st ed. Boston, MA, USA: Pearson Education, 2017.
4. Minteer, Analytics for the Internet of Things (IoT), 1st ed. Birmingham, UK: Packt Publishing, 2017.
5. H. Chaouchi, The Internet of Things: Connecting Objects to the Web, 1st ed. Hoboken, NJ, USA: Wiley, 2010.
6. P. Lea, Internet of Things for Architects, 1st ed. Birmingham, UK: Packt Publishing, 2018.
7. S. Pal, V. García Díaz, and R. P. Mohanty, IoT: Security and Privacy Paradigm, 1st ed. Boca Raton, FL, USA: CRC Press, 2022.

Reference books:

1. A. McEwen and H. Cassimally, Designing the Internet of Things, 1st ed. Chichester, UK: Wiley, 2014.
2. D. Norris, Raspberry Pi Projects for the Evil Genius, 2nd ed. New York, NY, USA: McGraw-Hill, 2014.
3. A. Tamboli, Build Your Own IoT Platform, 1st ed. Berkeley, CA, USA: Apress, 2019.

Online References:

1. Introduction To Internet Of Things SWAYAM Course:https://onlinecourses.nptel.ac.in/noc26_cs37/
2. Introduction to Internet of Things: Design Concept and Use cases SWAYAM Course:https://onlinecourses.swayam2.ac.in/ntr24_ed44/
3. Free IOT Course: https://www.simplilearn.com/learn-iot-basics-skillup
4. Coursera Course: https://www.coursera.org/learn/introduction-to-internet-of-things/
5. Introduction to the Internet of Things (IoT) MOOC Course: https://courses.mooc.fi/org/uh-cs/courses/introduction-to-the-internet-of-things-mooc

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question 1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question 3 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC5013	Ethical Hacking	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC5013	Ethical Hacking	20	20	60	--	--	100

Pre- requisite:
1. CEC402: Operating System

Program Outcomes Addressed
1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO5: Engineering Tool Usage
4. PO7: Ethics
5. PO11: Life-Long Learning

Course Objectives: The course aims to enable students:
1. To provide foundational knowledge of networking concepts, security basics, and ethical hacking principles.
2. To introduce techniques for gathering information using active and passive footprinting methods
3. To develop skills in network scanning, vulnerability scanning, and enumeration techniques.
4. To equip learners with methods of exploiting system vulnerabilities, privilege escalation, and maintaining system access
5. To provide understanding of web application threats, OWASP principles, and secure web design concepts
6. To introduce wireless network security concepts and techniques for wireless attacks and defenses.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Explain network fundamentals, security threats, and the phases of ethical hacking.
2. Interpret active and passive reconnaissance techniques for security assessment.
3. Apply scanning techniques and evaluate system information using enumeration tools
4. Analyze System vulnerabilities and execute controlled exploitation techniques.
5. Evaluate web vulnerabilities using OWASP guidelines and recommend secure configurations.
6. Design basic defensive strategies for wireless networks.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Networking & Ethical Hacking Basics**	**08**	
	1.1	Network Fundamentals, Network Topologies, Network Components, TCP/IP Networking Basics, TCP/IP Protocol Stack: DNS, SNMP, TCP, UDP, IP, ARP, RARP, ICMP protocols		**CO1**
	1.2	Review of the Security Basics: Attributes, Mechanisms and Attacks Taxonomy. The CIA Traid. Threats, Vulnerabilities, Attacks		
	1.3	Introduction to Ethical Hacking. Hacker Classifications: The Hats. Phases of Hacking.		
		Self-learning Topics: Linux Command-Line Essentials		
2.0		**Foot printing & Reconnaissance**	**06**	
	2.1	Introduction to Footprinting, Active & Passive Footprinting. Information Gathering Techniques		**CO2**
	2.2	Passive Information Gathering- Search Engine, Social Media & Public Profile, Email Footprinting.		
	2.3	Active Information Gathering- DNS, Network, Web Footprinting		
	2.4	Advanced Reconnaissance Techniques- Metadata extraction from public files, IoT device footprinting		
		Self-learning Topics: Cloud Footprinting		
3.0		**Scanning and Enumeration**	**06**	
	3.1	Scanning fundamentals, Types of scanning (Active and Passive)		**CO3**
	3.2	Network Scanning, Vulnerability Scanning, Identifying, Live Systems and Open Ports, Scanning Techniques: TCP/UDP Scans, SYN Scans, scanning tools		
	3.3	Enumeration Fundamentals, Enumeration techniques and tools, Enumeration Countermeasures		
		Self-learning Topics: Google Hacking (GHDB) and Doxing		
4.0		**System Hacking and Gaining Access**	**06**	
	4.1	Introduction to System Hacking, Exploiting System Vulnerabilities, Password Cracking Techniques (Brute Force, Dictionary Attack), Privilege Escalation and Maintaining Access.		**CO4**
	4.2	Malware: Keyloggers, Trojans, Rootkits		
	4.3	Session Management & Maintaining Access, Network-based Gaining Access Techniques, Exploiting System Misconfigurations		
		Self-learning Topics: Cloud system hacking fundamentals		
5.0		**Web Application Security and vulnerabilities**	**08**	
	5.1	Introduction to web applications security, threats and OWASP principles, OWASP top 10 web application vulnerabilities (SQL Injection, XSS, CSRF), introduction to secure design, web server: introduction a secure setup of apache, firewalling a server Browser: general concepts, functionalities, browsers war, configuration, and users tracking/profiling, browser security		**CO5**
	5.2	OWASP Privacy preserving, attacks to privacy, Tracking techniques, Advanced browser configuration, anonymity and onion routing (Tor).		

		Self-learning Topics: Advanced Injection Attacks (NoSQL Injection, XPath Injection, XXE)		
6.0		**Wireless Hacking Basics**	**05**	
	6.1	Fundamentals of Wireless Networks and Security, Wireless Hacking techniques, Cracking WEP/WPA/WPA2 Encryption, MITM (Man-in-the-Middle) Attacks in Wireless Networks		**CO6**
	6.2	Wireless Intrusion Detection & Prevention, Wireless Security Best Practices.		
		Self-learning Topics: Advanced Wi-Fi Attack Techniques- KRACK attack, Wi-Fi 6 & Wi-Fi 7 Security Enhancements		
		Total	**39**	

Textbooks:

1. J. Slavio, "Hacking: A Beginners' Guide to Computer Hacking, Basic Security and Penetration Testing", vol. 1. Scotts Valley, CA, USA: CreateSpace Independent Publishing Platform, 2016, pp. 1–84. ISBN: 978-1539937197.
2. Y. Diogenes and E. Ozkaya, "Cybersecurity Attack and Defense Strategies: Counter Modern Threats and Employ State-of-the-Art Tools and Techniques to Protect Your Organization Against Cybercriminals", 2nd ed. Birmingham, U.K.: Packt Publishing, 2019, pp. 1–634. ISBN: 978-1-83882-779-3.
3. R. Shimonski, "Cyber Reconnaissance, Surveillance and Defense", 1st ed. Waltham, MA, USA: Syngress, 2014, ISBN: 978-0124078140.

Reference books:

1. M. Sikorski and A. Honig, "Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software". San Francisco, CA, USA: No Starch Press, 2012, pp. 1–800. ISBN: 978-1593272906.
2. D. Stuttard and M. Pinto, "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws", 2nd ed. Indianapolis, IN, USA: Wiley Publishing, 2011, pp. 1–912. ISBN: 978-1118026472.

Online References:

1. NPTEL https://nptel.ac.in/courses/106105217
2. OWASP (Open Web Application Security Project) https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
3. Computer Security Student, LLC: Training Partnerships https://www.computersecuritystudent.com/

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question 1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question 3 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC5014	Data Warehouse and Mining	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC5014	Data Warehouse and Mining	20	20	60	--	--	100

Pre- requisite:

1. CEC304: Database management systems

Program Outcomes Addressed

1. PO1 Engineering Knowledge
2. PO2 Problem analysis
3. PO3 Design/Development of solutions
4. PO4 Conduct investigations of complex problems
5. PO5 Modern Tool Usage
6. PO11 Life-long Learning

Course Objectives: The course aims to enable students:

1. To identify the significance of Data Warehousing and Mining.
2. To develop understanding of various data preprocessing and data exploration techniques for knowledge discovery.
3. To analyze data and choose relevant classification algorithms for solving real-world problems.
4. To analyze data and choose relevant clustering algorithms for solving real-world problems.
5. To apply association rule mining in finding frequent patterns in various applications.
6. To apply web data mining in analyzing web structure.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Design data warehouse with dimensional modelling and apply OLAP operations.
2. Apply various data preprocessing and visualization techniques.
3. Classify real-world problems using appropriate classification algorithms.
4. Identify the need for clustering in real world scenarios and apply the same.
5. Analyze Association rules mining to identify frequent patterns.
6. Describe complex information and social networks with respect to web mining.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Datawarehouse fundamentals and Online Analytical Processing**	**08**	
	1.1	Introduction to Data Warehouse, Data warehouse architecture, Data warehouse versus Data Marts		**CO1**
	1.2	Data Warehouse Schemas; Star Schema, Snowflake Schema, Factless Fact Table, Fact Constellation Schema. Update to the dimension tables. Major steps in ETL process		
	1.3	OLAP operations: Slice, Dice, Rollup, Drilldown and Pivot.		
		Self-learning Topics: Cloud-based data warehouses		
2.0		**Introduction to Data Mining, Data Exploration and Data Pre-processing**	**08**	
	2.1	Data Mining Task Primitives, KDD process, Applications of Data Mining, Data Preprocessing: summarization, Cleaning, Data Integration & transformation		**CO2**
	2.2	Data reduction: Attribute subset selection, Histograms, Clustering and Sampling, Data Discretization: Normalization, Binning, Histogram Analysis and Concept hierarchy generation.		
	2.3	Statistical Description of Data, Data Exploration: Types of Attributes,		
	2.4	Proximity Measures for Nominal Attributes and Binary Attributes, Dissimilarity of Numeric Data		
		Self-learning Topics. Data Visualization, Explore BI tools.		
3.0		**Classification Basic Concepts**	**08**	
	3.1	Basic Concepts: Classification method, Decision Tree Induction, Attribute Selection Measures, ID3 algorithm		**CO3**
	3.2	Naïve Bayes classification, k-Nearest neighbor		
	3.3	Model evaluation and selection, Accuracy and Error measures, Precision, Recall, F1-score, specificity Confusion matrix, etc.		
	3.4	Evaluating the Accuracy of a Classifier: Holdout & Random Subsampling, Cross Validation, Bootstrap.		
		Self-learning Topics: Concept of Random Forest		
4.0		**Clustering Analysis**	**05**	
	4.1	Types of data in Cluster analysis, Partitioning Methods (k-Means, k-Medoids),		**CO4**
	4.2	Hierarchical Methods (Agglomerative, Divisive). Density based clustering: DBSCAN algorithm		
		Self-Learning: Advanced clustering algorithms BIRCH, CLIQUE		
5.0		**Association Rules mining**	**06**	
	5.1	Market Basket Analysis, Frequent Item sets, Closed Item sets, and		**CO5**

		Association Rule, Frequent Pattern Mining		
	5.2	Apriori Algorithm, Association Rule Generation, Improving the Efficiency of Apriori, Hash based technique, Transaction Reduction, ECLAT		
	5.3	Introduction to Mining Multilevel Association Rules and Mining Multidimensional Association Rules.		
		Self-learning Topics: Constraint-based mining		
6.0		**Web Mining**	**04**	
	6.1	Introduction, Web Content Mining: Crawlers, Personalization		**CO6**
	6.2	Web Structure Mining: Page Rank, HITS or Clever		
	6.3	Web Usage Mining.		
		Self-Learning: API-based data extraction		
		Total	**39**	

Textbooks:

1. Jiawei Han, Micheline Kamber, “Data mining: concepts and techniques", Morgan Kaufmann Publisher 2012, third edition, ISBN 978-0-12-381479-1
2. G. K. Gupta, “Introduction to Data mining with Case Studies", PHI Learning Private Limited, Delhi 2014, third edition, ISBN-978-81-203-5002-1.
3. William H Inmon, “Building the data Warehouse”, Wiley Publication 2005, fourth edition, ISBN: 978-0-764- 59944-6.

Reference books:

1. Reema Theraja, “Data warehousing”, Oxford University Press 2009.
2. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, “Introduction to Data Mining”, Pearson Publisher 2nd edition.
3. Ian H. Witten, Eibe Frank and Mark A. Hall, “Data Mining”, Morgan Kaufmann 3rd edition.

Online References:

1. Han and Camber PPT: https://hanj.cs.illinois.edu/bk3/bk3_slidesindex.htm
2. SWAYAM Course: https://onlinecourses.swayam2.ac.in/imb25_mg200/preview
3. Data Mining NPTEL Course: https://onlinecourses.nptel.ac.in/noc21_cs06/preview

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question 1 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question 3 (20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEL501	DevOps Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEL501	DevOps Lab	--	--	--	25	25	50

Pre- requisite:

1. CEL401: Operating System Lab
2. FEL205: Object Oriented Programming Methodology Lab

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design / Development of Solutions.
4. PO5: Engineering Tool Usage
5. PO7: Ethics
6. PO8: Individual and Collaborative Teamwork
7. PO9: Communication
8. PO11: Life-Long Learning

Lab Objectives: The course aims to enable students:

1. To provide hands-on experience with Linux operating system and shell scripting for automation of system-level tasks.
2. To enable students to use version control systems for managing and collaborating on software projects.
3. To develop skills in implementing Continuous Integration (CI) using modern DevOps tools.
4. To automate build, test, and deployment processes using industry-standard open-source tools.
5. To understand and apply containerization and orchestration technologies for application deployment.
6. To introduce configuration management, infrastructure automation, and monitoring using open-source DevOps tools.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Apply Linux commands and basic shell scripting to automate administrative and DevOps-related tasks.
2. Use Git-based version control systems to manage source code and support collaborative development.
3. Design and implement Continuous Integration pipelines using tools such as Jenkins and GitHub Actions.
4. Automate build, testing, and continuous deployment using Maven/Gradle, Selenium, and CI/CD pipelines.
5. Create, deploy, and manage containerized applications using Docker, Docker Compose, and Kubernetes.
6. Implement configuration management, infrastructure as code, and monitoring solutions using Ansible, Terraform, Prometheus, and Grafana.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	To install and configure a Linux environment and write basic shell scripts to automate simple system administration tasks (configure users, permission, SSH). **Task**: A mid-sized organization is setting up a new Linux server to manage its development team. Design and implement a solution by: Creating user accounts and assigning them to appropriate groups, setting file and directory permissions for secure access control, Configuring SSH for secure remote login (preferably key-based authentication), Write a shell script to automate these tasks for future scalability	LO1
2	To use Git and GitHub for source code management by performing repository creation, branching, merging, and collaboration operations. **Task**: A team of five developers is working on a web application but facing issues like code conflicts and difficulty tracking changes. To solve this, set up a GitHub repository and use feature branches so each developer can work independently. Merge changes using pull requests and follow proper commit practices to keep the work organized and collaborative.	LO2
3	To design and execute a Continuous Integration workflow using GitHub Actions to automatically build and test an application. **Task**: A development team is building an application and wants to automate code building and testing using GitHub Actions. Manual testing is slow and can lead to missed errors, affecting software quality, design a Continuous Integration workflow that automatically builds and tests the application on every code commit to ensure reliability.	LO3
4	To configure Jenkins and create CI pipelines for automated build and integration of application code. **Task**: A development team is facing delays and errors due to manual application builds. To improve efficiency, they need an automated solution for building and integrating code. As a DevOps engineer, set up Jenkins and create a pipeline to streamline the process and ensure consistent builds.	LO3
5	To automate the software build process using Maven or Gradle and generate deployable application artifacts. **Task**: A development team is facing problems because they are compiling and packaging applications manually, which leads to inconsistent builds. This often results in errors and delays during deployment. To improve this, automate the build process using Maven or Gradle to generate reliable JAR/WAR files.	LO4
6	To develop and execute automated web application test cases using Selenium WebDriver. **Task:** A QA team is testing a web application and aims to automate repetitive test cases using Selenium WebDriver. Manual testing is time-consuming and prone to human error, especially	LO4

	for regression testing, develop and execute automated test scripts to ensure application functionality, accuracy, and efficiency.	
7	To integrate Selenium test automation with Jenkins for continuous testing in a CI pipeline. **Task:** An organization uses a CI process where builds are generated automatically, but testing is performed separately, leading to late detection of defects and delayed releases. To improve software quality and ensure early bug detection, it is required to integrate automated testing into the CI pipeline by incorporating Selenium test cases within a Jenkins pipeline	LO4
8	To understand containerization concepts by running and managing applications inside Docker containers. **Task**: A development team wants to ensure consistent application deployment across different environments using Docker. Managing dependencies manually leads to compatibility issues and inconsistent performance across systems. The task is to understand containerization by running and managing applications inside Docker containers to achieve portability and efficiency.	LO5
9	To create Docker images using Dockerfiles and deploy containerized applications. **Task**: An application needs to be deployed across different environments, but variations in system configurations and dependencies result in inconsistent behavior. To ensure reliable and uniform deployment, it is necessary to adopt containerization by creating a Docker file and building a Docker image that packages the application along with its dependencies	LO5
10	To deploy and manage a multi-container application using Docker Compose **Task:** A development team is building an application with multiple services (such as frontend, backend, and database) and needs an efficient way to manage them together using Docker Compose. Manually running and linking containers is complex and error-prone, deploy and manage the multi-container application using Docker Compose to ensure seamless integration and simplified management.	LO5
11	To deploy, scale, and manage containerized applications using Kubernetes for orchestration. **Task**: In a production environment, lack of centralized monitoring makes it difficult to track server health, services, and network ports, leading to delayed issue detection. To ensure timely monitoring and alerts, implement a solution by installing and configuring Nagios for Linux and Windows systems.	LO5
12	To implement an end-to-end CI/CD pipeline using Jenkins and Docker for automated application deployment **Task:** A development team wants to automate the build, test, and deployment process of an application using Jenkins and Docker. Manual deployment is time-consuming and prone to errors, leading to delays and inconsistencies. Implement an end-to-end CI/CD pipeline to ensure faster, reliable, and automated application deployment.	LO5
13	To automate system configuration and application deployment using Ansible Playbooks.	LO6

	Task: In a multi-server environment, manual configuration often leads to inconsistencies and errors across systems. This makes management difficult and time-consuming. To ensure uniform configuration, automate the setup using Ansible Playbooks.	
14	To provision and manage infrastructure using Infrastructure as Code principles with Terraform. **Task:** A DevOps team wants to automate infrastructure provisioning and management using Terraform. Manual configuration of servers and resources is time-consuming and leads to inconsistencies across environments, implement Infrastructure as Code principles to provision and manage infrastructure efficiently and reliably.	LO6
15	To monitor application and system performance using Prometheus and visualize metrics using Grafana dashboards. **Task**: In the current system, there is no proper way to monitor performance in real time, making it hard to analyze system health. This affects decision-making and issue detection. To address this, implement monitoring using Prometheus and visualize metrics through Grafana dashboards.	LO6

Textbooks:

1. Paul Swartout, DevOps: Continuous Delivery, Integration, and Deployment with DevOps, Pearson Education, First Edition.
2. Mikael Krief, Learning DevOps – Continuous Delivery, Automation, and Cloud, Packt Publishing, First Edition.
3. Nigel Poulton, Docker Deep Dive, O'Reilly Media, First Edition.
4. Kelsey Hightower, Brendan Burns, Joe Beda, Kubernetes: Up and Running, O'Reilly Media, Second Edition.

Reference books:

1. Gene Kim, Jez Humble, Patrick Debois, John Willis, The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, IT Revolution Press, First Edition.
2. Jeff Geerling, Ansible for DevOps: Server and Configuration Management for Humans, O'Reilly Media, Second Edition.
3. Kief Morris, Infrastructure as Code: Managing Servers in the Cloud, O'Reilly Media, First Edition.
4. Jennifer Davis, Ryn Daniels, Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale, O'Reilly Media, First Edition.

Online References:

1. Git Documentation https://git-scm.com/doc
2. Jenkins Official Documentation https://www.jenkins.io/doc/
3. Docker Official Documentation https://docs.docker.com/
4. Kubernetes Official Documentation https://kubernetes.io/docs/
5. Ansible Official Documentation https://docs.ansible.com/

Software Tools

1. Operating System: Ubuntu Linux (Open Source)
2. Version Control System: Git, GitHub
3. CI/CD Tools: Jenkins, GitHub Actions
4. Build Tools: Maven / Gradle
5. Testing Tool: Selenium WebDriver

6. Containerization Tools: Docker, Docker Compose
7. Container Orchestration: Kubernetes (Minikube)
8. Configuration Management: Ansible
9. Infrastructure as Code: Terraform
10. Monitoring Tools: Prometheus, Grafana

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered.)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

- End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEL502	Computer Network Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEL502	Computer Network Lab	--	--	--	25	25	50

Pre- requisite:
1. FEL105: Engineering Workshop I

Program Outcomes Addressed
1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct investigations of complex problems
5. PO5: Engineering tool usage
6. PO8: Individual and Collaborative Team work
7. PO9: Communication
8. PO11: Lifelong Learning

Lab Objectives: The course aims to enable students:
1. To provide hands-on experience with Linux operating system and shell scripting for automation of system-level tasks.
2. To enable students to use version control systems for managing and collaborating on software projects.
3. To develop skills in implementing Continuous Integration (CI) using modern DevOps tools.
4. To automate build, test, and deployment processes using industry-standard open-source tools.
5. To understand and apply containerization and orchestration technologies for application deployment.
6. To introduce configuration management, infrastructure automation, and monitoring using open-source DevOps tools.

Lab Outcomes: Upon completion of this course, Students will be able to:
1. Simulate the functions of different networking devices and networking commands.
2. Examine the packet formats of each layer in TCP/IP model.
3. Interpret the functioning of MAC protocols of Data Link Layer.
4. Experiment socket programming on different applications.
5. Construct networks to implement network routing protocols.
6. Illustrate the working of Application Layer protocols

Suggested List of Experiments: Students are required to complete at least 12 experiments.		
Sr. No.	**Title of Experiments**	**LO Mapped**
1	A small manufacturing unit is setting up its internal network for communication between departments. They need to connect multiple systems, provide internet access, and ensure signal strength across floors. **Tasks:** • Identify which device (Router, Switch, Hub, Modem, Repeater) should be used at different points. • Justify the selection based on functionality and performance. • Explain how data flows through these devices in the given setup.	LO1
2	You are a network administrator troubleshooting connectivity issues in a Linux-based server system. **Tasks:** • Execute commands to check IP configuration, connectivity, and routing. • Diagnose why a system cannot reach an external server. • Suggest corrective actions using Linux networking commands.	LO1
3	A company is facing intermittent network issues and wants to analyze packet-level communication using Wireshark. **Tasks:** • Capture packets and analyze each TCP/IP layer: o Ethernet Layer: Identify frame header details and size. o Data Link Layer: Examine MAC addresses and ARP request/response. o Network Layer: Analyze IP headers, fragmentation, ICMP echo requests/replies. o Transport Layer: Identify TCP ports and 3-way handshake process. o Application Layer: Interpret DHCP, FTP, and HTTP headers. • Conclude where the delay or failure occurs in the communication.	LO2
4	An IT engineer notices that a system can ping an IP address but fails to resolve MAC addresses intermittently. **Tasks:** • Design a simple network. • Demonstrate PING (ICMP) functionality. • Analyze ARP table behavior and identify possible causes of failure.	LO2
5	A company is migrating from manual IP configuration to automated networking while maintaining control over routing and security. Tasks: a) Configure LAN with static and dynamic IP addressing. b) Using Linux: • View routing table • Add/delete routes • Change default gateway • Implement packet filtering using IP tables and enable IP forwarding. • Analyze how security rules affect packet flow.	LO1

6	A network experiences frequent collisions during data transmission in a shared medium environment. **Tasks:** • Simulate ALOHA and CSMA/CD protocols. • Compare efficiency and collision handling. • Recommend the best protocol for the scenario.	LO3
7	A company wants to separate departments (HR, Finance, Production) logically without changing physical infrastructure. **Tasks:** • Design VLAN-based network. • Configure VLANs and assign ports. • Demonstrate inter-VLAN communication.	LO5
8	A multi-branch organization requires manual control over routing paths between networks. **Tasks:** • Configure static routing. • Test connectivity between networks. • Analyze routing decisions.	LO5
9	A growing organization needs dynamic routing for scalability. **Tasks:** • Configure RIP protocol. • Observe routing table updates. • Analyze convergence behavior.	LO5
10	A network suffers from congestion due to heavy traffic load. **Tasks:** • Demonstrate: o Slow Start o Congestion Avoidance o Fast Retransmit o Fast Recovery • Visualize using Python/NS2. • Interpret how congestion is controlled.	LO5
11	An organization is granted a block of addresses with the beginning address 14.24.74.0/24. The organization needs to have 3 subblocks of addresses to use in its three subnets as shown below: • One subblock of 120 addresses. • One subblock of 60 addresses. One subblock of 10 addresses.	LO5
12	An organization is granted the block 130.34.12.64/26.The organization needs 4 subnets each with equal no hosts. Design the sub networks, find the information about each n/w and implement NAT.	LO5
13	A company wants to develop an internal chat system for communication between employees. **Tasks:** • Develop a client-server chat application using socket programming. • Demonstrate message exchange. • Handle multiple clients (optional advanced).	LO4

14	An organization wants automated IP allocation and centralized file sharing. **Tasks:** • Configure DHCP server. • Assign IP dynamically. • Perform file transfer using FTP. • Analyze security concerns.	LO6
15	An administrator needs to securely manage remote servers. **Tasks:** • Configure SSH access. • Perform remote login. • Compare SSH with Telnet in terms of security.	LO6

Textbooks:

1. Fourauzan B.,"Data Communications and Networking", 6th Edition, Tata McGraw-Hill, Publications, ISBN:0–07 – 058408 – 7
2. Andrew S. Tanenbaum, Computer Networks, 5th Edition, Pearson India, 2012.
3. Kurose, Ross, “Computer Networking a Top-Down Approach Featuring the Internet”, Pearson, ISBN-10: 0132856204
4. William Stallings, Data and Computer Communications, 10th Edition, Pearson Education, 2013.

Reference books:

1. L. Peterson and B. Davie, “Computer Networks: A Systems Approach”, 5th Edition, Morgan- Kaufmann, 2012.
2. Douglas E. Comer & M.S Narayanan,”Computer Network & Internet”, Pearson Education.
3. Nader F. Mir, Computer and Communication Networks, Second Edition, Prentice Hall, 2014.

Online References:

1. NPTEL- Computer Networks and Internet Protocol, IIT Kharagpur: nptel.ac.in/courses/106105183
2. NPTEL-Advanced Computer Networks: Advanced Computer Networks - Course

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered.)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

- End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEL503	Interpersonal and Career Skills	-	2*+2	-	-	02	-	02

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEL503	Interpersonal and Career Skills	-	-	-	50	-	50

Pre-requisite:

1. FEC205: Professional Communication Techniques

Program Outcome mapped:

1. PO6: The Engineer and the World
2. PO7: Ethics
3. PO8: Individual and Collaborative Teamwork
4. PO9: Communication
5. PO10: Project Management & Finance
6. PO11: Lifelong Learning

Course Objectives: The course aims to enable students:

1. To equip students with career development skills to align with the corporate and global requirements.
2. To develop students' ability to conduct and participate in professional business meetings.
3. To apply the knowledge of effective writing skills to produce technical/business documents.
4. To train students in creating effective, dynamic technical/business presentations.
5. To build strong interpersonal skills required for effective workplace communication.
6. To familiarise students with corporate ethics and Intellectual Property Rights.

Course Outcomes Upon completion of this course, Students will be able to:

1. Apply essential skills while appearing for recruitment processes or applying for higher education for a successful career.
2. Demonstrate professional behaviour; to plan, participate in, and document business meetings effectively.
3. Compose well-structured technical/business documents.
4. Design and deliver effective technical/business presentations using suitable strategies.
5. Employ strong interpersonal skills to overcome workplace challenges by understanding professional communication.
6. Practice principles of corporate ethics and understand Intellectual Property Rights.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.		**Career Skills**	**08**	
	1.1	SWOT Analysis		**CO1**
	1.2	Cover letter, Resume/CV:Resume Building, Types of Resumes (Chronological, Functional & Combination), Curriculum Vitae, Difference between Resume and CV		
	1.3	Group Discussion: Purpose, Types of GDs (Normal, Case-based & Role Plays)		
	1.4	GD Etiquettes, Parameters of Evaluating a GD		
	1.5	Interviews (Face to face and Virtual): Planning and Preparation, Types of Questions, Types of Interviews (Structured, Stress, Behavioural, Problem Solving & Case-based)		
	1.6	Modes of Interviews: Face-to-face, Telephonic		
	1.7	Virtual Interview planning and preparation, Online interview etiquettes		
	1.8	Statement of Purpose		
2.		**Business Meetings**	**03**	
	2.1	Documentation:Notice, Agenda, Minutes of the Meeting		**CO2**
	2.2	Conducting Business Meetings:Types of Meetings, Roles and Responsibilities of Chairperson, Secretary and Members, Meeting Etiquettes		
	2.3	Writing Minutes of the Meeting		
3.		**Advanced Technical Writing**	**06**	
	3.1	Purpose and Classification of Reports: Classification based on Subject Matter (Technology, Accounting, Finance, Marketing, etc.), Time Interval (Periodic, One-time, Special)		**CO3**
	3.2	Function (Informational, Analytical, etc.), Physical Factors (Memorandum, Letter, Short & Long)		
	3.3	Language and Style:Tense, Person & Voice of Reports, Numbering Style of Chapters, Sections and Figures/Tables, Proof-reading through Plagiarism Checkers		
	3.4	Project Report: Prefatory Parts (Front Matter), Report Proper (Main Body), Appended Parts (Back Matter)		
	3.5	Proposals: Format and Style of Technical Proposals, Parts of a Proposal		
	3.6	Technical Paper Writing: Parts of a Technical Paper, Language, Style and Formatting, Referencing in IEEE Format		
4.		**Effective Presentation**	**02**	
	4.1	Effective Presentation Strategies: Defining purpose, analysing audience & location, Structuring a presentation, Types of presentation aids, Using body language and Voice modulation		**CO4**
	4.2	Group Presentation: Sharing Responsibility in a Team, Building the contents and visuals together, Transition Phases		

5.		**Interpersonal Skills**	**05**	
	5.1	Emotional Intelligence		**CO5**
	5.2	Leadership		
	5.3	Time Management		
	5.4	Decision Making		
	5.5	Responsible Use of social media		
6.		**Corporate Ethics and Intellectual Property Rights**	**02**	
	6.1	Introduction to Corporate Ethics		**CO6**
	6.2	Intellectual Property Rights: Copyrights, Trademarks, Patents, Industrial Designs, Geographical Indications, Integrated Circuits, Trade Secrets (Undisclosed Information)		
		Total	**26**	

Reference books:

1. J. Butterfield, Professionalism : Soft Skills for a Digital Workplace. Boston, Massachusettes: Cengage Learning, © Course Technology, Cengage Learning, 2011.
2. R. V. Lesikar and J. D. Petit, Report Writing for Business, 10th ed. McGraw-Hill/Irwin, 1995.
3. L. A. Olsen and T. N. Huckin, Technical Writing and Professional Communication, 2nd ed. McGraw-Hill Humanities, Social Sciences & World Languages, 1991.
4. L Ann Masters and H. Wallace, Personal Development for Life and Work., 12th ed. Mason, Oh: South-Western Cengage Learning, 2011.
5. H. A. Murphy and Herbert William Hildebrandt, Effective Business Communications,
6. 7th ed. McGraw-Hill Companies, 1988.
7. F. Luthans, Organization Behaviour, 12th ed. New York: Mcgraw Hill, 2011.
8. R. C. Sharma and K. Mohan, Business Correspondence and Report Writing : A
9. Practical Approach to Business & Technical Communication, 6th ed. New Delhi: Tata
10. Mcgraw-Hill, 2017.
11. K. Alex, Soft skills : Know yourself & know the world, 3rd ed. New Delhi: S. Chand & Company Ltd, 2012.
12. R Subramanian, Professional ethics, 2nd ed. New Delhi: Oxford University Press, 2013.
13. S. Robbins and T. A. Judge, Organisational Behaviour, 18th ed. Pearson Higher
14. Education AU, 2024.

Online References:

1. **Directorate General of Training (DGT).**
Employability Skills Curriculum. Ministry of Skill Development & Entrepreneurship, Government of India, https://www.dgt.gov.in/sites/default/files/Employability_Skills_Curriculum.pdf.
2. **Punjab State Open University (PSOU).**
Technical Writing and IPR Syllabus. PSOU, https://psou.ac.in/asset/docs/course_syllabus/20221125183033165f0acdb4.pdf.
3. **National Institute of Technology Kurukshetra (NITKKR).**
Professional Ethics and Intellectual Property Rights (HSIR-13). NITKKR, https://nitkkr.ac.in/wp-content/uploads/2024/07/Professional-Ethics-and-IPR-HSIR-13.pdf.
4. **Commonwealth Corporation**.

Career Readiness Curriculum Guide. CommCorp, https://commcorp.org/wp-content/uploads/2016/07/resources_eyf-career-readiness-curriculum-guide.pdf.

Course Assessment:

- Term Work (CIAP): 50 Marks
- Term Work (CIAP) shall consist of Oral & Written assessment for 25 marks each
- Oral evaluation to be conducted for 25 Marks through:
  - Report Presentation 10 marks
  - Group Discussion 10 marks
  - Mock Interviews 05 marks
- Written evaluation to be conducted for 25 Marks through:
  - Project Report (25 to 30 pages) 10 marks
  - Assignments based on all modules 10 marks
  - Attendance 05 marks
- CIAP will be conducted based on the above scheme for 50 Marks.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDML5021	Data Analytics and Visualization Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
MDML5021	Data Analytics and Visualization Lab	--	--	--	25	--	25

Pre- requisite:
1. CEL 304: Skill Lab - Python Programming

Program Outcomes Addressed
1. PO1: Engineering Knowledge:
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct Investigations of Complex Problems
5. PO5: Engineering Tool Usage
6. PO9: Communication
7. PO11: Lifelong Learning

Lab Objectives: The course aims to enable students:
1. To demonstrate effective use of libraries in data analytics
2. To implement regression methods in real data analytics scenarios.
3. To use time series techniques to perform predictive analysis
4. To Implement text analytics techniques in practical scenarios
5. To use appropriate visualization methods in R and Python for data analysis.
6. To explore data visualization concepts using practical visualization tools.

Lab Outcomes: Upon completion of this course, Students will be able to:
1. Explore various data analytics Libraries in R and Python
2. Implement various Regression techniques for prediction.
3. Build various time series models on a given data set.
4. Design Text Analytics Application on a given data set.
5. Interpret various visualizations of given data sets using various tools
6. Develop interactive dashboards integrating data transformation, visualization for professional reporting.

Suggested List of Experiments: Students are required to complete at least 12 experiments.		
Sr. No.	**Title of Experiments**	**LO Mapped**
1	**Data Analytics Libraries Exploration** A retail company has collected large volumes of sales and customer data but lacks meaningful insights. As a data analyst, you are required to explore and utilize data analytics libraries in Python and R to process and visualize the data. **Tasks:** • Use Python libraries (NumPy, Pandas, Matplotlib) and R libraries (dplyr, tidyr, ggplot2). • Perform data cleaning and transformation. • Generate summary statistics. • Create basic visualizations to identify trends and patterns. **Recommended Datasets:** • https://www.kaggle.com/datasets/rohitsahoo/sales-forecasting • https://archive.ics.uci.edu/ml/datasets/Online+Retail	LO1
2	**Simple Linear Regression** A real estate firm wants to predict house prices based on property size (area). You are required to build a predictive model to understand this relationship. **Tasks:** • Implement simple linear regression in Python/R. • Visualize the regression line. • Evaluate model performance using R^2. • Interpret the relationship between variables. **Recommended Datasets:** • https://www.kaggle.com/datasets/harishkumardatalab/housing-price-prediction • https://archive.ics.uci.edu/ml/datasets/Housing	LO1
3	**Multiple Linear Regression** A marketing company wants to analyze how different advertising channels (TV, radio, digital) impact product sales. **Tasks:** • Build a multiple regression model. • Analyze the influence of each independent variable. • Perform feature selection. • Interpret coefficients and statistical significance. **Recommended Datasets:** • https://www.kaggle.com/datasets/ashydv/advertising-dataset • https://www.kaggle.com/datasets/sazid28/advertising.csv	LO2
4	**Time Series Analysis** A supermarket chain wants to analyze monthly sales data to identify trends and seasonal demand patterns for better inventory management. **Tasks:**	LO3

	• Perform time series decomposition. • Identify trend and seasonality components. • Apply rolling statistics. • Visualize temporal patterns. **Recommended Datasets:** • https://www.kaggle.com/datasets/kyanyoga/sample-sales-data • https://www.kaggle.com/datasets/robikscube/time-series-dataset	
5	**ARIMA Forecasting Model** An energy company wants to forecast future electricity demand based on historical consumption data. **Tasks:** • Identify ARIMA parameters (p, d, q). • Build and train the ARIMA model. • Perform forecasting. • Compare predicted vs actual values and evaluate accuracy. **Recommended Datasets:** • https://www.kaggle.com/datasets/robikscube/hourly-energy-consumption • https://www.kaggle.com/datasets/sudalairajkumar/daily-temperature-of-major-cities	LO4
6	**Geospatial Data Visualization** A city planning authority needs to visualize infrastructure distribution such as roads, buildings, and land areas for urban planning. **Tasks:** • Load and preprocess geospatial data (vector and raster). • Use tools like QGIS / GeoPandas / Rasterio. • Create spatial visualizations (maps). • Interpret spatial patterns. **Recommended Datasets:** • https://www.naturalearthdata.com/downloads/ • https://earthdata.nasa.gov/	LO5
7	**Geospatial Data Analytics** Perform supervised classification on Sentinel images to identify Land Use / Land Cover (Google Earth Engine). An environmental agency wants to monitor land use changes such as urban expansion, deforestation, and water bodies. **Tasks:** • Use Google Earth Engine. • Train supervised classification models. • Classify satellite images (Sentinel data). • Generate Land Use/Land Cover maps. **Recommended Datasets:** • https://developers.google.com/earth-engine/datasets • https://www.kaggle.com/datasets/apollo2506/satellite-imagery-datasets	LO5
8	**Text analytics** Implementation of Spam filter/Sentiment analysis in python/R.	LO4

	An e-commerce company wants to analyze customer reviews to understand satisfaction levels and detect spam messages. **Tasks:** • Preprocess text data (tokenization, stopword removal, etc.). • Build classification models. • Perform sentiment/spam detection. • Evaluate using confusion matrix, accuracy, precision, and recall. **Recommended Datasets:** • https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews • https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection	
9	**Visualization experiments in R using different Libraries.** A financial analyst needs to present stock market trends and investment insights to stakeholders using effective visualizations. **Tasks:** • Use R libraries (ggplot2, plotly, lattice). • Create at least two types of visualizations. • Ensure clarity and storytelling. • Interpret key insights. **Recommended Datasets:** • https://www.kaggle.com/datasets/borismarjanovic/price-volume-data-for-all-us-stocks-etfs • https://finance.yahoo.com/	LO5
10	**Visualization experiments in R using different Libraries.** A telecom company is experiencing high customer churn and wants to understand the underlying patterns. As a data analyst, you are provided with customer data including usage patterns, billing information, service complaints, and demographics. **Tasks:** • Use R libraries (ggplot2, plotly, lattice). • Create at least two types of visualizations. • Ensure clarity and storytelling. • Interpret key insights. **Recommended Datasets:** • https://www.kaggle.com/datasets/blastchar/telco-customer-churn	
11	**Visualization experiments in python using different Libraries.** A healthcare organization wants to analyze patient data to identify patterns in disease occurrence and improve treatment planning. **Tasks:** • Use Python libraries (Matplotlib, Seaborn, Plotly, Bokeh). • Create advanced visualizations (heatmaps, pair plots, interactive charts). • Identify trends and correlations. • Present actionable insights.	LO5

	Recommended Datasets: • https://www.kaggle.com/datasets/ronitf/heart-disease-uci • https://www.kaggle.com/datasets/imdevskp/corona-virus-report	
12	**Visualization experiments in python using different Libraries.** A manufacturing company wants to improve production efficiency by analyzing machine performance, downtime, defect rates, and output across different shifts and plants. **Tasks:** • Use Python libraries (Matplotlib, Seaborn, Plotly, Bokeh). • Create advanced visualizations such as time-series plots, heatmaps (machine vs downtime), and comparative bar charts. • Analyze patterns in machine breakdowns, defect rates, and production output. • Identify correlations between variables such as shift timing, machine usage, and defects. • Present actionable insights to reduce downtime and improve overall equipment effectiveness (OEE). **Recommended Datasets:** • https://www.kaggle.com/datasets/shivamb/machine-predictive-maintenance-classification	
13	**Build one complete interactive dashboard (Power BI/Tableau)** A business manager requires a real-time dashboard to monitor KPIs such as sales, profit, and regional performance. **Tasks:** • Use Power BI or Tableau. • Integrate multiple datasets. • Apply filters, slicers, and calculated fields (DAX). • Design interactive and user-friendly dashboards **Recommended Datasets:** • https://www.kaggle.com/datasets/kyanyoga/sample-sales-data • https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce	LO6
14	**Build one complete interactive dashboard (Power BI/Tableau)** A manufacturing company wants to monitor plant performance across multiple units, focusing on KPIs such as production output, downtime, equipment efficiency (OEE), and maintenance costs. The management needs a real-time dashboard to support operational and strategic decisions. **Tasks:** • Integrate multiple datasets (production logs, maintenance records, downtime reports). • Apply filters, slicers, and calculated fields (DAX) for KPIs like OEE, MTTR, and MTBF. • Design interactive dashboards with drill-down capabilities. • Identify bottlenecks, high downtime areas, and cost drivers. • Present actionable insights to improve efficiency and reduce operational losses. **Recommended Datasets:** • https://www.kaggle.com/datasets/arnabbiswas1/microsoft-azure-predictive-	

	maintenance	
15	To explore advanced chart types (waterfall, treemap funnel, map) in Power BI A logistics company wants to analyze supply chain efficiency and operational performance using advanced visual tools. **Tasks:** • Create charts like waterfall, treemap, funnel, and map visuals. • Integrate visuals into a dashboard. • Analyze hierarchical and flow-based data. • Provide insights for decision-making. **Recommended Datasets:** • https://www.kaggle.com/datasets/shashwatwork/dataco-smart-supply-chain-for-big-data-analysis • https://www.kaggle.com/datasets/ramakrishnanthiyagu/supply-chain-management	LO6

Textbooks:

1. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, Indianapolis, IN, USA: Wiley, 2015.
2. R. V. McCarthy, M. M. McCarthy, and W. Ceccucci, Applying Predictive Analytics: Finding Value in Data, 2nd ed., Cham, Switzerland: Springer, 2019.
3. N. D. Lewis, Introduction to Data Science: A Practical Approach with R and Python, Hoboken, NJ, USA: Wiley, 2017.
4. W. McKinney, Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter, 3rd ed., Sebastopol, CA, USA: O'Reilly Media, 2022.

Reference books:

1. B. Motwani, Data Analytics Using R, New Delhi, India: Wiley, 2015.
2. W. McKinney, Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter, 3rd ed., Sebastopol, CA, USA: O'Reilly Media, 2022.
3. J. Schwabish, Better Data Visualizations: A Guide for Scholars, Researchers, and Wonks, New York, NY, USA: Columbia University Press, 2021.
4. A. Albright and W. L. Winston, Business Analytics: Data Analysis and Decision Making, 7th ed., Boston, MA, USA: Cengage Learning, 2023.

Online Resources:

1. Coursera-Data Visualization with Python: Data Visualization with Python | Coursera
2. Microsoft-Fundamentals of data visualization with Power BI: Explore fundamentals of data visualization with Power BI | DP-900T00A-Azure-Data-Fundamentals
3. NPTEL course on R-Programming https://onlinecourses.nptel.ac.in/noc19_ma33/preview

Software Tools

1. Open-Source Data Visualization Tools

Term Work:

- Term work should consist of at least 12 experiments. (Ensure that all the LOs are covered)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDML5031	Embedded Systems and RTOS Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
MDML5031	Embedded Systems and RTOS Lab	--	--	--	25	--	25

Pre- requisite:

1. FEC204: Digital System Design
2. FEL103: C Programming Lab
3. MDMC4031: Microprocessors & Microcontrollers

Program Outcomes Addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO5: Engineering Tool Usage
5. PO6: The Engineer and The World
6. PO7: Ethics

Lab Objectives: The course aims to enable students:

1. Develop hands-on skills in configuring microcontrollers (8051/ARM/STM32/Arduino) using digital I/O, timers, ADC, and interrupts.
2. Interface sensors, actuators, and communication peripherals such as push buttons, LDRs, buzzers, motors, and I2C-based devices.
3. Implement low-power design techniques and evaluate power efficiency in embedded applications.
4. Apply embedded communication standards and protocols including UART, I2C, and interrupt-driven interfaces.
5. Apply functional safety and cybersecurity standards (IEC 61508, IEC 60601, ISO 26262, ISO 21434) through case study analysis.
6. Develop Real-Time Operating System (RTOS) based embedded applications, demonstrating multitasking, scheduling, and inter-process synchronization.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Interface and program basic embedded peripherals and sensors using GPIO, ADC, PWM, and interrupt logic.
2. Configure communication interfaces such as UART and I2C to connect external modules like DS1307 RTC.
3. Design and implement low-power embedded applications and analyze energy behavior.

4. Interpret and apply embedded safety and security requirements for compliance-driven design.
5. Develop RTOS-based applications using tasks, queues, events, and semaphores.
6. Implement and evaluate scheduling algorithms (RMS/EDF) to ensure deterministic real-time behavior.

Suggested List of Experiments:

Sr. No.	Title of Experiments	LO Mapped
1	Interfacing a Push Button with Interrupt Handling on an Embedded Core (8051/ ARM/ STM32) **Task:** Interface a push button to a GPIO pin on an embedded core (8051, ARM, or STM32), configure the pin for external interrupt (rising/falling edge), and handle the interrupt in an ISR to perform actions like toggling an LED or incrementing a counter.	LO1
2	Interfacing a Buzzer / Audio Alert Device with an Embedded Core (8051/ ARM/ STM32) **Task:** Interface a buzzer or audio alert device with an embedded core (8051, ARM, or STM32) to generate simple tones or alerts via GPIO toggling or PWM signals in a controlled experiment.	LO1
3	Interfacing an Analog Light Sensor (LDR) and Displaying Values on Serial Monitor **Task:** Interface an analog Light Dependent Resistor (LDR) to an ESP32 ADC pin, read real-time light intensity via voltage divider, convert raw ADC values to approximate lux or percentage, and display readings on the serial monitor using a FreeRTOS task.	LO2
4	Interfacing a simple Temperature Sensor and Displaying Real-Time ADC Values on Serial Monitor **Task:** Interface a simple analog temperature sensor (LM35) to an ESP32 GPIO pin, read real-time ADC values, convert to voltage and temperature, and display raw ADC, voltage, and temperature readings on the serial monitor using a FreeRTOS task.	LO2
5	Implement the I2C communication to connect to DS1307 RTC **Task:** Implement I2C communication with the DS1307 Real Time Clock (RTC) module to initialize the device, set the current date/time, and periodically read/display the time in a FreeRTOS-based embedded application.	LO2
6	Implementing a Simple Temperature-Controlled Fan System (Sensor + DC Motor) **Task:** Implement a temperature-controlled fan system using a DHT11 sensor to read ambient temperature and adjust DC motor (fan) speed via PWM based on thresholds (e.g., off below 25°C, variable speed above) in a FreeRTOS application on ESP32.	LO3
7	Implement power management in any embedded core of your choice **Task:** Implement power management on an ESP32 embedded core by configuring low-power modes (light sleep and deep sleep), using wake-up sources (timer, GPIO touch), measuring current reduction, and integrating with FreeRTOS tasks for periodic operation.	LO3

8	Interfacing a DC Motor (Speed and Direction Control) with Embedded Core **Task:** Interface a DC motor with an embedded microcontroller to control both speed (using PWM) and direction (using H-bridge logic) in a FreeRTOS-based application, demonstrating forward/reverse rotation and variable speeds.	LO3
9	Case Study Analysis and Compliance Mapping for IEC 61508, IEC 60601, and ISO 26262 **Task:** Analyze a hypothetical case study (e.g., development of an automotive-integrated medical monitoring device) for functional safety compliance, mapping requirements across IEC 61508 (general), IEC 60601 (medical devices), and ISO 26262 (automotive) to identify overlaps, differences, and verification strategies.	LO4
10	Developing a Safety Lifecycle Flow and Verification Checklist Based on IEC 61508 or ISO 26262 **Task:** Develop a safety lifecycle flow diagram outlining key phases (concept, product development at system/hardware/software levels, production, operation, and decommissioning) based on IEC 61508 or ISO 26262 and create a verification checklist for compliance in each phase.	LO4
11	Porting of FreeRTOS to Arduino/STM32 **Task:** Port FreeRTOS to Arduino (AVR-based Uno) and STM32 platforms by the appropriate FreeRTOS port, configuring the kernel, creating basic tasks, and verifying multitasking behavior through LED blinking or serial output.	LO5
12	Write a Program to illustrate the Queue Management Features of FreeRTOS **Task:** Develop a C program in FreeRTOS to demonstrate queue management by creating a producer task that sends structured data items to a queue and multiple consumer tasks that receive and process items from the queue, illustrating blocking, non-blocking, and timeout behaviors.	LO5
13	Write a Program to illustrate the Event Management Features of FreeRTOS **Task:** Develop a C program in FreeRTOS to demonstrate event group features by creating multiple tasks that set specific event bits upon certain conditions and a monitoring task that waits for combinations of events (using logical AND/OR) to synchronize actions.	LO5
14	Write a Program to illustrate the use of Binary and Counting Semaphore for Task Synchronisation using FreeRTOS **Task:** Develop a C program in FreeRTOS demonstrating task synchronization with a binary semaphore (for one-to-one signaling between producer and consumer tasks) and a counting semaphore (for managing multiple resource instances or events in a producer-consumer scenario).	LO5
15	Write a Program to Create Multiple Tasks and understand the Multitasking	LO6

	capabilities of RTOS(FreeRTOS) **Task:** Create multiple independent tasks in FreeRTOS with different priorities and delays to demonstrate preemptive multitasking, context switching, and concurrent execution on a single-core microcontroller.	

Textbooks:

1. K. C. S. Murti, Design Principles for Embedded Systems. Boca Raton, FL, USA: CRC Press, 2017.
2. C. Hobbs, Embedded Software Development for Safety-Critical Systems, 1st ed. Oxford, U.K.: Newnes (Elsevier), 2015.
3. B. Amos, Hands-On RTOS with Microcontrollers, 1st ed. Birmingham, U.K.: Packt Publishing, 2020.

Reference books:

1. M. J. Pont, The Engineering of Reliable Embedded Systems, 2nd ed. London, U.K.: Routledge, 2017.
2. P. Marwedel, Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems and IoT, 4th ed. Cham, Switzerland: Springer, 2021.

Online References:

1. ControllersTech Tutorials – https://controllerstech.com
2. FreeRTOS Official Documentation & Tutorials – https://www.freertos.org
3. Learn Embedded Systems GitHub Repository – https://github.com/iam-sandipmaity/learn-embedded-systems

Software Tools:

1. STM32CubeIDE / Keil / GCC Toolchain
2. FreeRTOS Kernel
3. MCU Debuggers / On-chip Debug Tools
4. Logic Analyzer (Hardware/Software)
5. Serial Terminal / Data Visualization Tools (PuTTY / TeraTerm / Arduino Serial Monitor)
6. Simulation Software (Proteus)

Term Work:

- Term work should consist of at least 10 experiments+ 2 experiments on simulator (Ensure that all the LOs are covered)
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) = 20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL5011	Advance Database Management System Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL5011	Advance Database Management System Lab	--	--	--	25	-	25

Pre-requisite:

1. CEL302 - Database Management System Lab

Program Outcomes:

1. PO1: Engineering Knowledge.
2. PO3: Design/development of solutions.
3. PO5: Engineering Tool Usage.
4. PO8: Individual and Collaborative Teamwork.
5. PO9: Communication.
6. PO11: Life-long learning.

Lab Objectives: The course aims to enable students:

1. To understand advanced database concepts through practical applications.
2. To design and implement Enhanced Entity-Relationship (EER) models.
3. To explore distributed database techniques like fragmentation.
4. To analyze and estimate query costs for efficient database operations.
5. To gain hands-on experience with NoSQL databases like MongoDB.
6. To explore database security and connectivity with front-end systems.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Create and implement EER models for real-world scenarios.
2. Analyze distributed database fragmentation and query optimization.
3. Demonstrate secure database access using PostgreSQL.
4. Apply querying techniques on XML, JSON, and MongoDB databases for real-world data storage and retrieval.
5. Implement the concept of graph databases and differentiate them from relational and NoSQL databases.
6. Illustrate database connectivity to front-end applications and perform operations.

Suggested List of Experiments:

Sr. No.	Title of Experiments	LO Mapped
1	Design an EER model for a multinational company's database, including entities like employees, departments, and projects. Implement the model using SQL/PostgreSQL to manage data such as employee details, project assignments, and department information efficiently.	LO1
2	Analyze and implement horizontal and vertical fragmentation for the company database, which includes project, department, and employee data. Optimize data storage by fragmenting data based on location and department across multiple distributed sites.	LO2
3	A hospital database system is experiencing slow query responses due to large datasets. Apply techniques like indexing, cost-based optimization, and query plan analysis to improve performance in SQL/PostgreSQL. Focus on reducing query response time and ensuring efficient data retrieval for large datasets.	LO2
4	Design and implement security mechanisms such as DAC, MAC, and RBAC in a PostgreSQL database to control access to sensitive information based on user roles, security policies, and privileges. The system should support multi-level access control for real-world scenarios such as banking or healthcare applications.	LO3
5	Develop XML DTD and XML Schema files to validate customer information submitted through forms on an e-commerce website.	LO4
6	Use XPath and XQuery based on the FLWOR expression to extract product details (such as product name, category, price, and reviews) from an XML-based product catalog.	LO4
7	Develop a client-server communication system that parses and stringifies JSON data to exchange real-time stock market data between a server and a client. Implement operations to retrieve the latest stock prices, volumes, and other relevant data to ensure seamless information exchange.	LO4
8	Install and configure the client and server for MongoDB. Set up MongoDB as the backend for a real-time chat application. Design a schema for efficient message storage and retrieval.	LO4
9	Create a MongoDB database for an e-commerce platform, including collections for users, products, orders, and reviews. They will perform CRUD operations to manage data and use the find() method to query product details, user orders, and reviews efficiently.	LO4
10	Design and optimize queries for an e-commerce platform database using MongoDB. Perform advanced queries such as filtering products with logical operators, sorting by price, rating, and inventory status, and aggregating data to calculate total sales, average ratings, and identify trending products.	LO4
11	Design and deploy a graph database for a real-world application such as a social network, using Neo4j's Cypher query language. Focus on relationships between users, posts, and interactions, and how to efficiently query them.	LO5
12	Design a graph schema, perform queries using Cypher to identify key relationships, and discover patterns in graph data, such as social networks, recommendation systems, or organizational structures.	LO5
13	To write NOSQL QUERIES to understand the concept of open-source database Management systems such as CASSANDRA.	LO5
14	Case Study and Comparative Analysis of Key–Value NoSQL Databases Using Redis and DynamoDB.	LO5

15	Design and implement NoSQL database connectivity with a front-end application to perform CRUD operations and manage real-time data exchange for dynamic user interactions.	LO6

Textbooks:

1. Korth, Siberchatz, Sudarshan, "Database System Concepts", 7th Edition, McGraw-Hill ,2020-22
2. Elmasri and Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson Education ,2017.
3. Ozsu, M. Tamer, Valduriez, Patrick, "Principles of distributed database systems", 3rd Edition, Pearson Education, Inc. ,2020.
4. Pramod Sadalge, Martin Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, 1st Edition, Addison Wesley/Pearson ,2012.
5. Jeff Friesen, Java, XML and JSON, Second Edition,2019, après, Inc.

Reference Books:

1. Chhanda Ray, Distributed Database System, Pearson Education, India.
2. Adam Fowler, NoSQL for dummies, John Wiley & Sons, Inc.
3. Shashank Tiwari, Professional NOSQL, John Willy & Sons. Inc.

Online References:

1. W3Schools:- https://www.w3schools.com/xml/
2. JavaScriptObjectNotation(JSON):-https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Scripting/JSON
3. Neo4j Native graph database :- https://neo4j.com/docs/getting-started/cypher-intro/
4. MongoDB :- https://www.mongodb.com/docs/manual/

Software Tools

1. PostgreSQL:https://www.enterprisedb.com/downloads/postgres-postgresql-downloads
2. MySQL: https://dev.mysql.com/downloads/
3. MongoDB: https://www.mongodb.com/docs/manual/
4. Neo4j: https://neo4j.com/product/community-edition/

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered.
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as a Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance).
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL5012	Internet of Things Lab	--	2	--	--	1	--	1

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL5012	Internet of Things Lab	--	--	--	25	--	25

Pre- requisite:

1. FEC104: C-Programming
2. CEL304: Skill Lab - Python Programming
3. CEC305: Computer Organization and Architecture

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design / Development of Solutions.
4. PO5: Engineering Tool Usage
5. PO7: Ethics
6. PO8: Individual and Collaborative Teamwork
7. PO9: Communication
8. PO11: Life-Long Learning

Lab Objectives: The course aims to enable students:

1. To introduce the fundamental setup of IoT development environments, device configuration, and basic connectivity workflows.
2. To familiarize with sensor–actuator integration, data acquisition techniques, and hardware software interfacing.
3. To observe and interpret IoT data streams using monitoring, visualization, and debugging tools.
4. To expose communication challenges and security considerations in IoT through controlled experimentation with protocols and networks.
5. To use standard IoT platforms and frameworks for device management, data handling, and cloud integration.
6. To enhance understanding of IoT communication mechanisms through practical implementation of wireless technologies and protocol-based interactions.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Understand the basic IOT component usage in various applications.
2. Apply the concepts of functional blocks and components in the IoT.
3. Implement data calibration techniques for various sensors.
4. Evaluate sensor-generated data using suitable analytical methods.
5. Analyze various data handling methods in IoT-based systems using appropriate tools or platforms

6. Demonstrate various communication protocols in IoT with respect to their features, advantages, and limitations in different application scenarios.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	A smart agriculture company is developing an IoT-based solution to monitor soil moisture, temperature, and humidity in remote farms. They are evaluating different IoT development boards such as Arduino Uno, Raspberry Pi, and ESP8266. Based on the application requirements (low power consumption, wireless connectivity, and cost-effectiveness), classify these boards and compare their features. Suggest the most suitable board with justification.	LO1
2	**Simulation based:** A greenhouse owner wants to monitor environmental conditions to protect plants. **Task:** Study and demonstrate working of different sensors (temperature, humidity, light) and actuators (fan, buzzer), Arduino UNO,Temperature Sensor (TMP36 or simulated input), Light Sensor (LDR), Humidity (simulate using potentiometer), LED (as fan indicator), Buzzer (**Use Tinkercad)**	LO1
3	**Simulation based:** Develop a program to blink 5 LEDs back and forth. **Problem:** A festival organizer wants attractive lighting patterns for decoration. **Task:** Design a system where 5 LEDs glow in a running/back-and-forth pattern using Arduino UNO, 5 LEDs, 5 Resistors (220Ω), Breadboard, Jumper wires.	LO2
4	Develop a program to interface a relay with Arduino board. **Problem:** A homeowner wants to control high-voltage appliances safely using a microcontroller. **Task:** Interface a relay with Arduino to control a bulb/fan.	LO2
5	Develop a program to control a DC motor with Arduino board. **Problem:** A small room requires automated fan control for comfort. **Task:** Control a DC motor using Arduino (ON/OFF or speed control using PWM).	LO2
6	Develop a program to read the pH value of a various substances like milk, lime and water. **Problem:** A dairy company wants to check milk quality before distribution. **Task:** Measure pH values of milk, lime water, and normal water.	LO3
7	Develop a water level depth detection system using Ultrasonic sensor. **Problem:** Water overflow and shortage are common in residential buildings. **Task:** Use an Ultrasonic sensor to detect water level.	LO4
8	Implement Wi-Fi Connectivity on a Microcontroller to Enable Internet Access **Problem:** A company wants devices connected to the internet for remote monitoring. **Task:** Enable Wi-Fi connectivity using ESP8266/ESP32.	LO5
9	Develop a real-time interface for ultrasonic sensor data collection and push data on the cloud	LO5

	platform like Thingspeak. **Problem:** A building manager wants remote monitoring of tank levels. **Task:** Send Ultrasonic sensor data to ThingSpeak.	
10	Develop a real-time interface for DHT11 sensor data collection and push data on the cloud platform like Thingspeak **Problem:** Farmers need live temperature and humidity data. **Task:** Use DHT11 sensor and upload data to ThingSpeak.	LO5
11	To study various cloud platforms and application protocol like Thingspeak, Blynk **Problem:** A startup wants to choose the best IoT platform. **Task:** Study and compare: • ThingSpeak • Blynk	LO5
12	Develop a system to collect sensor data and push it to Google Sheets for analysis and logging. **Problem:** A school lab wants to maintain sensor records automatically. **Task:** Send sensor data to Google Sheets.	LO6
13	Develop a real-time interface for Monitoring Temperature and Humidity on web-based control platform and visualize data on a mobile interface using Application like Blynk. **Problem:** Users want to monitor weather from their mobile phones. **Task:** Display temperature & humidity on Blynk.	LO6
14	Develop a real-time interface to blink LEDs and demonstrate remote device control using the Blynk mobile application in an IoT setup. **Problem:** A user wants to control home lights from anywhere. **Task:** Control LED remotely using Blynk.	LO6
15	Demonstrate remote control of a relay using the MQTT communication Protocol in a real-time IoT scenario. **Problem:** A smart home system should control multiple devices. **Task:** Control LEDs/relay via mobile app.	LO6

Textbooks:

1. A. Bahga and V. Madisetti, Internet of Things: A Hands-on Approach. Hyderabad, India: University Press, 2015.
2. R. Kamal, Internet of Things: Architecture and Design Principles. New Delhi, India: McGraw Hill Education, 2017.
3. D. Hanes, G. Salgueiro, et al., IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things. Boston, MA, USA: Pearson Education, 2017.
4. A. Minteer, Analytics for the Internet of Things (IoT). Birmingham, UK: Packt Publishing, 2017.

Reference books:

1. McEwen and H. Cassimally, Designing the Internet of Things. Chichester, UK: John Wiley & Sons Inc., 2013.
2. Y. Kanetkar and S. Korde, 21 Internet of Things (IoT) Experiments. New Delhi, India: BPB Publications, 2019.

Online References:

1. Introduction To Internet of Things NPTEL Course: https://onlinecourses.nptel.ac.in/noc26_cs37/
2. Introduction to Internet of Things: Design Concept and Use case: https://onlinecourses.swayam2.ac.in/ntr24_ed44/
3. Free IoT Certification Course: https://www.simplilearn.com/learn-iot-basics-skillup
4. Introduction to Internet of Things: https://www.coursera.org/learn/introduction-to-internet-of-things/
5. MOOC Course: https://courses.mooc.fi/org/uh-cs/courses/introduction-to-the-internet-of-things-mooc

Software and Hardware Tools

A) Hardware Tools

- Arduino UNO / NodeMCU / ESP32 development boards
- Sensors
  - Ultrasonic sensor (HC-SR04)
  - DHT11 / DHT22 Temperature–Humidity sensor
  - LDR (Light Dependent Resistor)
  - Moisture / Soil sensor
  - pH sensor module
  - Gas sensor (MQ-2/MQ-135)
  - Sound sensor
- Actuators
  - LEDs
  - Relay module
  - DC motor + motor driver (L293D/L298N)
- Other Hardware
  - Breadboard
  - Jumper wires (male–male, male–female)
  - Resistors (220Ω, 1kΩ)
  - USB cables
  - Power supply/5V adapter

B) Software Tools

- Arduino IDE
- Python 3.x (optional for data processing)
- ESP Home / Platform IO (optional)
- MQTT Broker (Mosquitto or HiveMQ Web Client)
- Firebase / Google App Script (for Sheets integration)

Term Work:

- Term work shall consist of **10 Hardware and 2 Software simulation experiments**. (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment)+ 05 Marks (Attendance)
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL5013	Ethical Hacking Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL5013	Ethical Hacking Lab	--	--	--	25		25

Pre- requisite:

1. CEC402: Operating System

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO5: Engineering Tool Usage
4. PO7: Ethics
5. PO11: Life-Long Learning

Lab Objectives: The course aims to enable students:

1. To introduce the fundamental setup of virtualized ethical-hacking environments and basic reconnaissance tools.
2. To familiarize with active footprinting, DNS enumeration, and network-scanning methodologies.
3. To observe and understand network traffic using packet-analysis tools.
4. To expose web-application vulnerabilities through controlled exploitation practices.
5. To use standard security frameworks for vulnerability assessment and penetration testing.
6. To enhance understanding of authentication and wireless-security mechanisms through practical cracking techniques.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Apply virtualization and Linux environments to set up ethical hacking platforms.
2. Demonstrate reconnaissance, scanning, and enumeration using security tools.
3. Analyze network traffic and vulnerabilities using packet capture and assessment techniques.
4. Exploit SQL Injection, XSS, and other web vulnerabilities on controlled test platforms.
5. Implement password cracking, wireless attacks, and basic malware creation in a secure setup.
6. Perform CTF challenges to assess and strengthen system security.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	A cybersecurity analyst is tasked with setting up a secure penetration testing lab to perform reconnaissance on target systems. **Tasks:** • Construct a virtualized environment using VirtualBox and Kali Linux. • Perform Reconnaissance using tools such as Dig, Whois, Traceroute, and Ifconfig. • Analyze the information gathered to understand target network structure.	LO1
2	An organization wants to assess its external exposure by gathering detailed domain-related information from publicly available sources. **Tasks:** • Demonstrate active footprinting techniques on a target domain. • Perform DNS enumeration using DIG and NSLOOKUP tools. • Extract and analyze DNS records such as A, MX, NS, and CNAME. • Identify potential security risks based on the collected domain information.	LO2
3	A network security engineer is assigned to identify active hosts and analyze services running on a network to assess potential security risks. **Tasks:** • Perform network scanning using Nmap for host discovery. • Identify active hosts available in the network. • Analyze open ports and running services on target systems. • Interpret scan results to identify potential vulnerabilities and security risks.	LO2
4	A network administrator suspects unusual network activity and needs to monitor and analyze packet-level communication to identify potential issues. **Tasks:** • Capture network traffic using Wireshark. • Monitor and analyze packets across different layers of the network. • Identify suspicious activities or anomalies in the captured traffic. • Prepare a report summarizing the observed network behavior and findings.	LO3
5	A web application is suspected to contain vulnerabilities due to improper input validation, potentially allowing unauthorized access to the database. **Tasks:** • Investigate SQL Injection vulnerabilities in Damn Vulnerable Web Application (DVWA). • Analyze the behavior of SQL queries based on different user inputs. • Observe application responses to identify successful and failed injection attempts. • Evaluate the impact of SQL Injection on data security and system integrity. • Suggest appropriate mitigation techniques to prevent SQL Injection attacks.	LO4
6	A web application security analyst aims to identify client-side vulnerabilities caused by improper input validation that may lead to unauthorized script execution in users' browsers. **Tasks:**	LO4

	• Execute Cross-Site Scripting (XSS) attacks on bWAPP and Damn Vulnerable Web Application (DVWA). • Test different types of XSS attacks such as reflected and stored XSS. • Analyze how malicious scripts are executed in the client's browser. • Observe application behavior and identify vulnerable input fields. • Recommend appropriate mitigation techniques to prevent XSS vulnerabilities.	
7	A security analyst is required to evaluate vulnerabilities identified in a system to determine their severity and potential impact on organizational security. **Tasks:** • Perform vulnerability assessment using OpenVAS. • Analyze the identified vulnerabilities based on risk ratings. • Evaluate exploitability and potential impact of each vulnerability. • Prioritize vulnerabilities based on severity levels. • Recommend appropriate remediation strategies to mitigate identified risks.	LO4
8	A security analyst needs to validate and correlate results from automated vulnerability scans with manual verification to identify system misconfigurations and outdated services. **Tasks:** • Perform automated vulnerability scanning on a target system. • Manually verify the findings obtained from the scan. • Identify misconfigurations and outdated services present in the system. • Correlate automated and manual findings to eliminate false positives. • Analyze the impact of identified vulnerabilities on system security.	LO4
9	A web security analyst is tasked with systematically identifying vulnerabilities in a web application using structured testing methodologies. **Tasks:** • Set up and configure Burp Suite for web application testing. • Intercept and analyze HTTP/HTTPS requests and responses. • Identify vulnerabilities such as input validation flaws, authentication issues, and session management weaknesses. • Perform manual testing to validate identified vulnerabilities. • Document findings and recommend mitigation strategies.	LO4
10	A penetration tester is assigned to identify and validate system vulnerabilities using exploitation frameworks in a controlled environment. **Tasks:** • Set up and configure Metasploit Framework. • Perform vulnerability scanning and enumeration. • Execute suitable exploits to validate identified vulnerabilities. • Analyze system behavior post-exploitation. • Recommend remediation techniques to secure the system.	LO4
11	A cybersecurity researcher aims to study malware behavior in a controlled lab environment to understand its impact on system security. **Tasks:** • Develop a controlled malware simulation using Python.	LO5

	• Implement basic functionalities such as keystroke capture and persistence. • Execute the simulation in a secure and isolated environment. • Analyze behavior patterns and system impact. • Suggest preventive and detection mechanisms for such threats.	
12	A security analyst is required to evaluate different password cracking techniques to understand their effectiveness against various password strengths. **Tasks:** • Perform password cracking using John the Ripper and Hashcat. • Implement dictionary, brute-force, and hybrid attack techniques. • Compare effectiveness based on time and success rate. • Analyze the impact of password complexity on security. • Recommend best practices for strong password policies.	LO5
13	A network security engineer is tasked with assessing wireless network security to identify vulnerabilities and potential threats. **Tasks:** • Set up a wireless testing environment using Aircrack-ng. • Perform wireless network scanning and monitoring. • Identify available networks and security protocols used. • Analyze vulnerabilities in wireless configurations. • Recommend security measures to protect wireless networks.	LO5
14	A penetration tester needs to evaluate the security of a wireless network by capturing authentication handshakes and testing password strength. **Tasks:** • Capture WPA/WPA2 handshake using Aircrack-ng tools. • Analyze captured packets for authentication details. • Perform password recovery using appropriate techniques. • Evaluate the strength of the wireless network password. • Suggest improvements to enhance wireless security.	LO5
15	A cybersecurity student is required to apply foundational security techniques to solve Capture The Flag (CTF) challenges involving real-world scenarios. **Tasks:** • Perform information gathering and reconnaissance on given targets. • Identify vulnerabilities using appropriate tools and techniques. • Exploit identified weaknesses to retrieve hidden flags. • Document the methodology and steps followed. • Analyze the challenges and propose improved security measures.	LO6

Textbooks:

1. V. Ramachandran and C. Buchanan, “Kali Linux Wireless Penetration Testing Beginner's Guide”, 2nd ed. Birmingham, UK: Packt Publishing, 2015, ISBN: 978-1-78328-041-4
2. G. Khawaja, “Practical Web Penetration Testing: Secure Web Applications Using Burp Suite, Nmap, Metasploit, and More”. Birmingham, UK: Packt Publishing, 2018, ISBN: 978-1-78862-448-0.

Reference books:

1. P. Prasad, "Mastering Modern Web Penetration Testing." Birmingham, UK: Packt Publishing, Oct. 2016, ISBN: 978-1-78646-564-1.
2. W. Halton and B. Weaver, Kali Linux 2: "Windows Penetration Testing." Birmingham, UK: Packt Publishing, Jun. 2016, ISBN: 978-1-78646-162-9.
3. J. Clarke-Salt, "SQL Injection Attacks and Defense", 1st ed. Burlington, MA, USA: Syngress, 2009, ISBN: 978-1-59749-424-3.

Online References:

1. OWASP (Open Web Application Security Project) - https://owasp.org/
2. NIST (National Institute of Standards and Technology) - https://www.nist.gov/
3. Penetration Testing Execution Standard (PTES) - http://www.pentest-standard.org/
4. SANS Institute - https://www.sans.org/
5. Metasploit Unleashed - https://www.metasploitunleashed.com/
6. CERT (Computer Emergency Response Team) - https://www.cert.org/

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL5014	Data Warehouse and Mining Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL5014	Data Warehouse and Mining Lab	--	--	--	25	--	25

Pre- requisite:
1. FEC104 C-Programming
2. FEL205 Object Oriented Programming Methodology Lab
3. CEL304 Skill Lab - Python Programming

Program Outcomes Addressed
1. PO1 Engineering Knowledge
2. PO2 Problem Analysis
3. PO3 Design/Development of solutions
4. PO4 Conduct investigations of complex problems
5. PO5 Modern Tool Usage

Lab Objectives: The course aims to enable students:
1. Apply data warehouse design principles to construct basic warehouse schemas.
2. Implement data preprocessing techniques to clean and prepare real-time data.
3. Apply suitable classification algorithms using real-time datasets.
4. Apply clustering algorithms to analyze patterns within datasets.
5. Examine frequent pattern mining results to identify significant associations
6. Use WEKA software to perform data mining tasks including classification, clustering, and pattern mining.

Lab Outcomes: Upon completion of this course, Students will be able to:
1. Design data warehouses and perform various OLAP operations.
2. Analyze the data and apply various preprocessing techniques to clean the data.
3. Apply appropriate classification algorithms to any real-time data.
4. Apply appropriate clustering algorithms to any real-time data.
5. Apply frequent pattern matching algorithms to find useful patterns.
6. Apply web structure mining algorithm and use of WEKA tool.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO
1	A retail company stores sales data in multiple files, making analysis difficult. Management needs a centralized system for better decision-making. Design a data warehouse using star and snowflake schemas. Identify fact and dimension tables. Perform ETL using Power Query in Excel.	LO1
2	Create reports using Excel Power Pivot. Perform Slice, Dice, Roll-up, and Drill-down operations on retail company data warehouse (refer experiment no. 1). Use Pivot tables for summarization. Help management analyze trends efficiently.	LO1
3	A healthcare analytics company analyzes patient records to identify diabetes risk. The dataset includes features like glucose, BMI, age, insulin, and blood pressure, with possible missing values and outliers. Students must perform data preprocessing and EDA by handling null values, applying normalization and transformation, conducting descriptive analysis, and detecting/treating outliers using a Python notebook or visualization tool.	LO2
4	A retail company wants to better understand its customers for targeted marketing. The dataset contains attributes such as age, income, spending score, and purchase frequency. Apply discretization techniques like binning. Create visualizations such as histograms or charts. Extract useful insights.	LO2
5	HR Analytics helps us with interpreting organizational data. It finds out the people-related trends in the data and helps the HR Department take the appropriate steps to keep the organization running smoothly and profitably. Explore and analyze company's employees dataset to predict potential attrition cases, thereby helping the appropriate HR Personnel take the necessary steps to retain the employee.	LO2
6	A hospital wants to predict diseases using patient data. Historical data is available for training. Patient attributes used for prediction, such as: Age, Blood Pressure, Glucose Level, BMI, Insulin, Cholesterol (optional). Implement the KNN algorithm. Classify new patient records based on neighbors. Evaluate model accuracy.	LO3
7	An e-commerce platform wants to predict customer purchases. User behavior data is available.User profile data(User ID,Age,Gender Location) etc. Browsing Behavior (Number of pages visited,Time spent on website/app,Number of product views,Categories viewed,Search queries count) Implement any classification algorithm using Java or Python. Train and test the model. Measure performance.	LO3
8	An email system wants to detect spam messages. Use labeled data for training. Implement Naïve Bayes or Decision Tree algorithm. Classify emails as spam or not spam. Improve accuracy.	LO3
9	Perform data Pre-processing task and demonstrate Classification, Clustering, Association algorithm on data sets using data mining tool (WEKA/R tool)	LO3
10	A telecom company wants to segment its customers to design targeted marketing strategies. The company collects customer usage data such as monthly bill amount, data usage, call duration, and number of recharge plans. Students are required to apply the K-Means or K-Medoid clustering algorithm to group customers into similar segments and analyze their behavior patterns.	LO4

11	A research lab is analyzing biological samples to identify groups with similar characteristics. Each sample is described by features such as protein expression levels, gene activity, or chemical composition. The relationships between samples are important, so the lab uses hierarchical clustering to understand how samples group together. Students are required to apply hierarchical clustering (agglomerative method), generate a dendrogram, and identify meaningful clusters.	LO4
12	A research team is analyzing spatial data where clusters are irregular in shape and contain noise. Traditional clustering methods like K-Means fail to detect these patterns effectively. Students must apply DBSCAN to identify dense regions and noise points	LO4
13	A supermarket wants to analyze purchase patterns. Identify frequently bought item sets. Apply Apriori algorithm. Generate association rules. Improve sales strategies.	LO5
14	A search engine needs to rank web pages. Ranking depends on links between pages. Implement PageRank algorithm. Calculate page importance scores. Display relevant pages first.	LO6
15	A company collects customer feedback from reviews and social media. The data is unstructured and difficult to analyze manually. Apply text preprocessing techniques like tokenization and stopword removal. Implement sentiment analysis or text classification. Extract insights for business decisions.	LO6

Textbooks:

1. Jiawei Han, Micheline Kamber, “Data mining: concepts and techniques", Morgan Kaufmann Publisher 2012, third edition, ISBN 978-0-12-381479-1
2. G. K. Gupta, “Introduction to Data mining with Case Studies", PHI Learning Private Limited, Delhi 2014, third edition, ISBN-978-81-203-5002-1.
3. William H Inmon, “Building the data Warehouse”, Wiley Publication 2005, fourth edition, ISBN: 978-0-764- 59944-6.

Reference Books:

1. Dunham, M. H., “Data mining: Introductory and advanced topics”, Upper Saddle River, N.J: Pearson education/Prentice Hall 2003.
2. Ralph Kimball, Margy Ross, “The Data Warehouse Toolkit”, 3rd Edition, Wiley 2013, ISBN-13: 978- 11185308 01.
3. Giuseppe Bonaccorso, “Machine Learning Algorithms”, Packt Publishing Ltd. 2017, ISBN 978-178588-962- 2

Online References:

1. Data Management-Systems-Jiawei-Han-Micheline-Kamber-Jian-Pei-Data-Mining. Concepts-and Techniques-3rd-Edition-Morgan-Kaufmann-2011.pdf http://myweb.sabanciuniv.edu/rdehkharghani/files/2016/02/The-Morgan-Kaufmann-Series
2. Data Mining Weka_2nd_Ed_2005.pdf http://www.academia.dk/BiologiskAntropologi/Epidemiologi/DataMining/Witten_and_Frank
3. GitHub Pages documentation http://scikit-learn.org/stable/datasets/
4. Metrics and scoring: quantifying the quality of predictions https://scikit-learn.org/stable/modules/model_evaluation.html
5. Datasets https://www.kaggle.com/datasets

Software Tools

1. Google Colaboratory (Colab) — https://developers.google.com/colab
2. WEKA (machine-learning/data-mining tool) — https://ml.cms.waikato.ac.nz/weka/
3. Microsoft Power BI — https://www.microsoft.com/en-in/power-platform/products/power-bi/

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Credits Assigned			
		Theory	Practical	Tutorial	Total
CEM501	Mini Project 2	--	1	--	1

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEM501	Mini Project 2	--	--	--	25	25	50

Program Outcomes addressed:

1. PO1: Engineering knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.
7. PO7: Ethics.
8. PO8: Individual and Collaborative Team work.
9. PO9: Communication.
10. PO10: Project Management and Finance.
11. PO11: Life-Long Learning.

Project Objectives: The course aims to enable students:

1. To identify user or societal needs and formulate clear problem statements.
2. To work collaboratively for effective problem-solving.
3. To apply engineering fundamentals to develop feasible solutions.
4. To promote self-learning and research-oriented thinking.

Project Outcomes: Upon completion of this course, Students will be able to:

1. Identify and define real-world or research problems through requirement gathering and stakeholder interaction.
2. Develop project objectives, scope, and feasibility using software engineering and SDLC principles.
3. Analyze requirements and develop system designs using standard modelling techniques.
4. Implement solutions using appropriate technologies, engineering practices, and version control.
5. Test and validate the system through structured test cases, datasets, and simulations.
6. Compose technical documentation, present project outcomes, and demonstrate teamwork, ethics, and self-learning.

1 Guidelines for Mini Project

1.1 General Guidelines

a) Students shall form groups of 3 to 4 members; groups with fewer than three or more than four students are not permitted.
b) Students should conduct surveys to identify needs and develop problem statements in consultation with faculty.
c) Each group shall submit an implementation plan using a Gantt Chart, PERT, or CPM chart, covering weekly activities and milestones.
d) Every group must maintain a logbook to record weekly progress, tasks completed, challenges faced, and supervisor's comments and signatures.
e) Faculty supervisors may provide necessary guidance; however, emphasis shall be on self-learning, independent thinking, and initiative**.**
f) Students should understand the problem thoroughly, propose multiple solution approaches, and select the most appropriate solution in consultation with the supervisor.
g) Students shall develop the selected solution into a working model using domain-specific tools, technologies, or components and demonstrate the same.
h) The solution must be validated using proper justification through test cases, benchmark data, experimentation, or simulations, and the final report must be compiled as per the guidelines for the project report.
i) Students must follow ethical practices**,** maintain academic integrity, avoid plagiarism, and appropriately acknowledge all references and sources used.
j) Students are encouraged to use standard engineering tools such as version control systems (e.g., Git), project management tools, and proper documentation tools during project execution.
k) Evaluation will consider problem identification, planning, implementation quality, testing and validation, documentation, presentation, teamwork, and overall contribution.
l) Students should be encouraged to explore opportunities for innovation, entrepreneurship**,** and potential intellectual property creation such as patent, copyright, or research publication.
m) Students must follow lab safety norms and responsibly use equipment, hardware, and computational resources during the execution of the project.

1.2 Guidelines for the Project Report:

a) Report should not exceed 30 pages. Simply staple it to discourage use of plastic.
b) Report must contain Gantt chart, architectural diagram, Data Flow Diagrams, applicable UML diagrams, screenshot of outputs (Include only required information pages).
c) The recommended report writing format is in LaTeX.

2 Guidelines for Assessment of Mini Project:

2.1 Evaluation of Term Work (CIAP):

A. A Review/Progress Monitoring Committee shall be constituted by the Head of Department. Mini Project progress shall be evaluated continuously, with a minimum of two reviews in the semester.
B. Continuous assessment should also emphasize individual performance, including each student's contribution, understanding of the project, and responses during review interactions.
C. The Review/Progress Monitoring Committee may consider the following points:
    a) students' group shall complete project in all aspects including,
        i. Identification of need/problem
        ii. Proposed final solution
        iii. Procurement of components/systems

iv. Building prototype and testing

b) Two reviews will be conducted for continuous assessment,
   i. **First Review:** Finalization of problem and proposed solution
   ii. **Second Review:** Implementation and testing of the solution

D. The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

1.2 Distribution of Marks

The distribution of Term Work marks shall be as follows:

a) Marks by Guide/Supervisor based on Logbook: 10
b) Marks by Review Committee: 10
c) Quality of Project Report: 05

1.3 Evaluation of Rubrics CIAP (Term work)

Mini Project shall be assessed based on the following criteria;

a) Quality of Survey and Need Identification
b) Clarity and Completeness of Problem Definition
c) Innovativeness and Creativity in Proposed Solutions
d) Feasibility of Solution and Justification of Selected Approach
e) Cost Effectiveness and Societal/Environmental Impact
f) Quality of System Design and Use of Engineering Norms
g) Implementation Quality and Functioning of Prototype as per Requirements
h) Effective Use of Technical Skill Sets and Tools
i) Teamwork, Leadership, and Individual Contribution
j) Clarity in Documentation, Presentation, and Communication

All criteria in generic may be considered for evaluation of performance of students in mini project.

2.4 ESEP (Practical / Oral) Examination:

a) The report should be prepared as per the guidelines issued by the University of Mumbai.
b) Mini Project shall be assessed through a presentation and demonstration of working model by the student project group to a panel of Internal and External Examiners preferably from industry or research organization's having experience of more than five years approved by the head of Institution.
c) Students shall be motivated to publish a paper based on the work in Conferences/students competitions.
d) End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

2.5 Evaluation Rubrics ESEP (Practical / Oral)

Mini Project shall be assessed based on the following rubrics:

a) Quality of problem identification & clarity
b) Innovation & creativity in solution
c) Feasibility, cost effectiveness & societal impact
d) Functionality and correctness of the working model
e) Effective use of engineering tools and skill sets

f) Adherence to engineering standards and norms
g) Individual contribution and teamwork
h) Written and oral communication skills

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC601	System Programming and Compiler Construction	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEC601	System Programming and Compiler Construction	20	20	60	--	--	100

Pre-requisite:
1. CSC501: Theoretical Computer Science
2. CEC402: Operating System.
3. CEC305: Computer Organization and Architecture

Program Outcome mapped:
1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of solutions
4. PO7: Ethics
5. PO11: Life-Long Learning

Course Objectives: The course aims to enable students:
1. To introduce role and functionality of various system programs over application programs.
2. To explore basic concepts, structure and design of assemblers, macro processors, linkers and loaders.
3. To explore the need to follow the syntax in writing an application program.
4. To investigate the analysis phase of compiler designed to understand the programmer‘s requirements without ambiguity.
5. To synthesize the outcomes of analysis phase to produce the object code that is efficient in terms of space and execution time, by applying code optimization techniques.
6. To understand the working of parallel compilers.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Identify the relevance of different system programs and understand various data structures used for design of assembler.
2. Discuss microprocessor design and distinguish between different loaders and linkers and their contribution in developing efficient user applications.
3. Analyse lexical, Syntax and semantic analysis phases of compiler.
4. Generate Intermediate code as part of synthesis phase.

5. Illustrate Code optimizations, code generation and error detection.
6. Apply parallelization in compiler in sorting arrays.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1		**Introduction to System Programing and Assembler design**	**07**	
	1.1	Concept of System Software, Goals of system software, system program and system programming,		**CO1**
	1.2	Introduction to various system programs		
	1.3	Elements of Assembly Language programming, Assembly scheme, pass structure of assembler,		
	1.4	Assembler Design: Two pass assembler X86 processor, data structures used		
		Self-Learning Topics: X86 Addressing Modes & Instruction Format, Relocation and Linking Concepts		
2		**Macros, Linker and Loader**	**06**	
	2.1	Introduction, Macro definition and call, Design of Two pass macro processor, data structures used		**CO2**
	2.2	Introduction, functions of loaders, Relocation and Linking concept, Different loading schemes		
	2.3	Direct Linking Loader, Dynamic linking and loading		
		Self-Learning Topics: Advanced Macro Expansion Techniques, Loader implementation issues		
3		**Introduction to Compilers and Analysis Phase:**	**11**	
	3.1	Overview of compilers, difference between compilers and interpreters, structure of a compiler		**CO3**
	3.2	Role of lexical analyser, Input buffering, specification and recognition of tokens		
	3.3	Context-free grammars, Parsing techniques: Top Down and Bottom-Up approach, Recursive Descent Parser , Predictive Parser, LL(1) Parser.		
	3.4	LR Parsers - SLR Parser - Canonical LR Parser - LALR Parser. Operator Precedence Parser		
	3.5	Syntax-directed definitions, evaluation orders, and type checking		
		Self-Learning Topics: Ambiguity in grammars and ambiguity removal, Attribute Grammars (Synthesized & Inherited Attributes),		
4		**Intermediate Code Generation:**	**05**	
	4.1	Intermediate Code, Intermediate code generation methods		**CO4**
	4.2	Three Address Code representation		
	4.3	**Self-Learning Topics:** 3AC generation for loops and conditionals		
5		**Code optimization and Code generation**	**06**	

	5.1	Code Optimization: Sources of optimization, Loop Optimization: Unrolling, Fusion, Invariant Code Motion, optimization of basic blocks, loops, dataflow analysis ,Directed Acyclic Graph		**CO5**
	5.2	Design of a code generator, run-time storage management, target machine architecture.		
	5.3	Error Detection and Recovery: Lexical, syntactic, and semantic errors; error recovery strategies.		
		Self-Learning Topics: RISC vs CISC and impact on code generation		
6		**Parallel Compiler:**	**04**	
	6.1	Parallel compilation, parallel parsing, semantic analysis, and optimization techniques		**CO6**
	6.2	Parallel Code Generation: Code generation for parallel architectures, synchronization, instruction scheduling.		
		Self-Learning Topics: Loop parallelization techniques, Dependence analysis for parallel compilation		
		Total	**39**	

Textbooks:

1. M Dhamdhere: Systems programming and Operating Systems, Tata McGraw Hill, Second Edition,I SBN-13, 978-0074630839
2. A. V. Aho, R. Shethi, Monica Lam, J.D. Ulman: Compilers Principles, Techniques and Tools, Pearson Education, Updated Second Edition, ISBN 978-81317-2101-8, 2023
3. Modern Compiler Design, by Dick Grune, Kees van Reeuwijk, Henri E. Bal, Ceriel J. H. Jacobs, Koen Langendoen, 2nd ed, ISBN 978 1461 446989, Springer (2012).
4. Allen. “Optimizing Compilers for Modern Architectures.” (2004).

Reference Books:

1. Manoj Chandak and Khushboo Khurana”Compiler Design”(2018)
2. Randy Allen & Ken Kennedy, “Optimizing Compilers for Modern Architectures”, ISBN 81-8147-366-3, Morgan Kaufmann, Elsevier, 2002

Online References:

1. NPTEL Compiler Design Course: https://nptel.ac.in/courses/106105190
2. Project Centered Coursera Course” Build a Modern Computer from First Principles: Nand to Tetris Part II”:https://www.coursera.org/lecture/nand2tetris2/unit-4-1-syntax-analysis-5pC2Z

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC602	Cryptography and Network Security	03	-	-	03	-	-	03

Corse Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEC602	Cryptography and Network Security	20	20	60	--	--	100

Pre-requisite:
1. CEC303: Discrete Structure and Graph Theory
2. CEC503: Computer Networks

Program Outcome mapped:
1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of solutions
4. PO6: The Engineer and The World
5. PO7: Ethics
6. PO11: Life-Long Learning

Course Objectives: The course aims to enable students:
1. A basic understanding of security concepts, classical cryptographic techniques, and the need for secure communication.
2. Knowledge of symmetric key cryptography, mathematical foundations, and modern block cipher algorithms.
3. Knowledge of asymmetric key cryptography, major public-key algorithms, and key management mechanisms.
4. Understanding of cryptographic hash functions, digital certificates, and authentication protocols.
5. Insight into access control mechanisms, formal security models, and Network Access Control (NAC) implementations.
6. The ability to analyze network security threats, DoS attacks, and Internet security protocols with practical relevance.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Apply basic security principles and classical cryptographic techniques to secure communication.
2. Explain and apply mathematical foundations and symmetric key cryptographic algorithms.
3. Apply asymmetric key cryptographic techniques and key management mechanisms.
4. Analyze cryptographic hash functions, digital certificates, and authentication protocols.
5. Evaluate access control models and assess NAC enforcement techniques.
6. Analyze network security threats, DoS attack mechanisms, and major Internet security protocols.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1		**Introduction to Security Concepts and Basic Cryptography**	**07**	
	1.1	Security Goals, Services, Mechanisms and Types of Security Attacks		**CO1**
	1.2	The OSI security architecture, Network security model		
	1.3	Concept of Cryptography, Classical Encryption techniques: mono-alphabetic and poly-alphabetic		
	1.4	Substitution Techniques: Caesar Cipher, Multiplicative Cipher, Affine Cipher, Autokey Cipher, Vigenère Cipher, Playfair Cipher, Hill Cipher		
	1.5	Transposition Techniques: Keyed and Keyless Transposition Ciphers, Introduction to Steganography		
		Self-Learning Topics: Homomorphic Encryption, Basics of Quantum Cryptography		
2		**Symmetric Key Cryptography**	**07**	
	2.1	Euclidean algorithm, Modular Arithmetic, Prime Numbers, Fermat's Theorem, Congruence Relation, Chinese Remainder Theorem		**CO2**
	2.2	Block Cipher Principles, Block Cipher Modes of Operation, DES, Double DES, Triple DES, Advanced Encryption Standard (AES), Stream Ciphers: RC5 algorithm		
		Self-Learning Topics: Basics of Linear cryptanalysis and Differential Cryptanalysis		
3		**Asymmetric Key Cryptography and Key Management**	**07**	
	3.1	Public key cryptography, Principles of public key cryptosystems, The RSA algorithm, The Knapsack algorithm, ElGamal Algorithm		**CO3**
	3.2	Key management Techniques: Using Symmetric and Asymmetric Algorithms and Trusted Third Party, Diffie Hellman Key exchange Algorithm		
		Self-Learning Topics: Elliptic Curve Cryptography (ECC), Basics of Lattice-based Cryptography		
4		**Cryptographic Hash Functions, Digital Certificates and Authentication Protocols**	**09**	
	4.1	Cryptographic Hash Functions, Properties of Secure Hash Function, MAC, HMAC, CMAC, MD5, SHA-256 (SHA-2 family) and SHA-3 (Keccak)		**CO4**
	4.2	Digital Certificate: X.509, PKI, Digital Signature Schemes: RSA, EIGamal		
	4.3	User Authentication and Entity Authentication, Needham Schroeder Authentication Protocol, Kerberos Authentication Protocol		
		Self-Learning Topics: SHA-512, Schnorr Signature Schemes		
5		**Access Control**	**03**	
	5.1	Access Control Lists & Access Control Models: Bell–LaPadula Model, Biba Model, Clark–Wilson Model		**CO5**

	5.2	NAC: Principle Elements of NAC, Principle NAC Enforcement Methods, How to Implement NAC Solutions		
		Self-Learning Topics: Basics of Covert Channels		
6		**Network Security**	**06**	
	6.1	Network security basics: TCP/IP vulnerabilities (Layer wise), Packet Sniffing, ARP spoofing, port scanning, IP spoofing, TCP SYN flood, DNS Spoofing		**CO6**
	6.2	Denial of Service: Classic DOS attacks, Source Address spoofing, ICMP flood, SYN flood, UDP flood		
	6.3	Internet Security Protocols: SSL, IPSEC, Secure Email: PGP, Firewalls, IDS and types		
		Self-Learning Topics: Types of malware, DoS Mitigation Techniques, Virtual Private Network (VPN)		
		Total	**39**	

Textbooks:

1. W. Stallings, Cryptography and Network Security: Principles and Practice, 8th ed. Boston, MA, USA: Pearson, 2022.
2. E. Cole, Network Security Bible, 2nd ed. Hoboken, NJ, USA: Wiley, 2011.
3. S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Upper Saddle River, NJ, USA: Pearson Education, 2021.
4. S. Rajasekaran and G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications, 2nd ed. New Delhi, India: PHI Learning, 2017.

Reference books:

1. M. Gregg, Build Your Own Security Lab: A Field Guide for Network Testing, 1st ed. Indianapolis, IN, USA: Wiley, 2008.
2. T. Boyles, CCNA Security Study Guide: Exam 640-553, 1st ed. Hoboken, NJ, USA: Wiley/Sybex, 2010.
3. E. Rich and K. Knight, Artificial Intelligence, 5th ed. New Delhi, India: Tata McGraw-Hill Education, 2025.
4. M. T. Hagan, H. B. Demuth, and M. H. Beale, Neural Network Design, 2nd ed. Boston, MA, USA: Cengage Learning, 2014.

Online References

1. Geeks for Geeks- Comprehensive Cryptography Tutorial with Concepts and Algorithms: https://www.geeksforgeeks.org/cryptography/
2. NPTEL- Cryptography and Network Security: https://onlinecourses.nptel.ac.in/noc22_cs90/preview
3. Coursera – Cryptography I: https://www.coursera.org/learn/crypto
4. Coursera – Mathematical Foundations of Cryptography: https://www.coursera.org/learn/mathematical-foundations-cryptography
5. Tutorials Point – Cryptography: https://www.tutorialspoint.com/cryptography
6. Cyber Security Guide: https://cybersecurityguide.org/

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC603	Artificial Intelligence and Soft Computing	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEC603	Artificial Intelligence and Soft Computing	20	20	60	--	--	100

Pre- requisite:
1. CEC302: Data Structures
2. CEC403: Analysis of Algorithms

Program Outcomes Addressed
1. PO1 – Engineering Knowledge
2. PO2 – Problem Analysis
3. PO3 – Design/Development of Solutions
4. PO4 – Conduct Investigations of Complex Problems
5. PO9 – Communication
6. PO11- Lifelong Learning

Course Objectives: The course aims to enable students:
1. To understand fundamental concepts of artificial intelligence and soft computing.
2. To formulate a problem statement and gain knowledge of search techniques.
3. To understand knowledge representation and planning.
4. To understand fundamentals of Neural Networks and architecture.
5. To design Fuzzy Controllers.
6. To familiarize with evolutionary algorithms.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Identify the various characteristics of Artificial Intelligence and Soft Computing techniques.
2. Choose an appropriate problem-solving method for an agent to find a sequence of actions to reach the goal state.
3. Apply knowledge representation for reasoning.
4. Construct supervised and unsupervised ANN for real world applications.
5. Design fuzzy controller system.
6. Demonstrate working of evolutionary algorithms.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Artificial Intelligence (AI) and Soft Computing**	**04**	
	1.1	Definition of Artificial Intelligence, History of AI Intelligent Agents: Agents and Environments, Rationality, Nature of Environment, Structure of Agent, types of Agents, PEAS Properties		**CO1**
	1.2	Soft Computing: Introduction of soft computing, Randomness, Vagueness, Approximation, Uncertainty, soft computing vs. hard computing, various types of soft computing techniques		
	1.3	Applications of AI, Applications of Soft Computing		
		Self-learning Topics: Human–AI Interaction Basics, Introduction to Artificial Neural Networks (ANN) and perceptron		
2.0		**Problem Solving and Search Techniques**	**08**	
	2.1	Problem Formulation		**CO2**
	2.2	Uninformed Search Methods: Depth Limited Search, Depth First Iterative Deepening (DFID), Informed Search Method: Hill climbing Search, A* Search		
	2.3	Constraint Satisfaction Problem Solving, Water Jug Problem, Zero-Sum Game: Min-Max Algorithm, Alpha-Beta Pruning		
		Self-learning Topics: Memory Bound Algorithm: SMA, SMA*		
3.0		**Knowledge Representation, Reasoning and Planning**	**06**	
	3.1	Knowledge based agents		**CO3**
	3.2	First order logic: syntax and Semantic, Knowledge Engineering in FOL, Inference in FOL: Unification, Forward Chaining, Backward Chaining and Resolution		
	3.3	Partial Order Planning, Hierarchical Planning and Conditional Planning		
		Self-learning Topics: Bayesian Belief Network, Types of Planning		
4.0		**Artificial Neural Network**	**10**	
	4.1	Introduction – Biological Neural System, Elements of Artificial Neural Networks, McCulloch-Pitts Neuron, Activation Functions: Unipolar and Bipolar Functions		**CO4**
	4.2	Learning Rules: Hebbian Learning, Perceptron, Delta learning rule, Competitive Learning		
	4.3	Neural Network Architecture: Single layer Feed Forward ANN, Multilayer Feed Forward ANN, Back Propagation algorithm		
	4.4	Un-Supervised Learning algorithm: Kohonen Self Organizing Maps, Learning VQ		
		Self-learning Topics: Feedback Neural Network, Recurrent Neural Network, Radial Basis Network		
5.0		**Fuzzy System**	**07**	

	5.1	Introduction to Fuzzy Sets: Fuzzy set theory, Fuzzy set versus crisp set, fuzzy relations, membership functions: Triangular, Trapezoid, Sigmoid, Gaussian		**CO5**
	5.2	Fuzzy Logic: Fuzzy Logic basics, Fuzzy Rules and Fuzzy Reasoning		
	5.3	Fuzzy inference systems: Fuzzification Methods, Defuzzification Methods: Centre of Gravity, Mean-of-Max (MoM), Smallest of Max, Largest of Max		
	5.4	Mamdani Model for fuzzy controllers		
		Self-learning Topics: Takagi-Sugeno Model		
6.0		**Evolutionary Algorithms**	**04**	
	6.1	Introduction: Need for Evolutionary Algorithms, Difference of Classical Optimization and Evolutionary Algorithms		**CO6**
	6.2	Genetic algorithm: Selection Operators, Cross Over Operators, Elitism, Exploration and Exploitation Tradeoff		
	6.3	Simulated annealing, Particle Swarm Optimization		
		Self-learning Topics: Expert system: Introduction, Characteristics, Architecture, Stages in the development of expert system		
		Total	**39**	

Textbooks:

1. Stuart J. Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach”, 4th Edition, Pearson Education, 2020.
2. S. N. Sivanandam and S. N. Deepa, “Principles of Soft Computing”, 3rd Edition, Wiley India, 2018.
3. S. Rajasekaran and G. A. Vijayalakshmi Pai, “Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications”, 2nd Edition, PHI Learning, 2017.
4. N. P. Padhy, “Artificial Intelligence and Intelligent Systems”, 1st Edition, Oxford University Press, 2018.

Reference books:

1. Elaine Rich and Kevin Knight, "Artificial Intelligence", 5th Edition, Tata McGraw-Hill Education, 2025.
2. Satish Kumar, "Neural Networks: A Classroom Approach", 2nd Edition, Tata McGraw-Hill Education, 2017.
3. Martin T. Hagan, Howard B. Demuth, and Mark H. Beale, "Neural Network Design", 2nd Edition, Cengage Learning, 2014.
4. J.-S. R. Jang, C.-T. Sun, and E. Mizutani, "Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence", 1st Edition, PHI Learning, 2003.
5. Jacek M. Zurada, "Introduction to Artificial Neural Systems", 1st Edition, Jaico Publishing House, 1992.
6. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", 4th Edition, Wiley, 2021.

Online References:

1. An Introduction to Artificial Intelligence – https://share.google/umcQLMykKscKcgb5k
2. Artificial Intelligence: Concepts and Techniques –https://share.google/8JeKlR2eOnVXNdBVt
3. Introduction To Soft Computing - Course https://share.google/Gyex84j4McZQQw3zI
4. Introduction To Soft Computing –NPTEL+ https://share.google/FWefkZDqu5BIHkiCn

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC6023	Decision Making and Business Intelligence	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE[$]			
		ISE	MSE				
MDMC6023	Decision Making and Business Intelligence	20	20	60	--	--	100

Pre-requisite:
1. CEC304: Database management systems
2. MDMC5022: Data Analytics & Visualization

Program Outcomes Addressed
1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solutions
4. PO4: Investigation of Complex problems
5. PO5: Engineering Tool usage

Course Objectives: The course aims to enable students:
1. Understand the fundamentals of decision-making processes.
2. Analyze different types of decisions and decision environments.
3. Explain the architecture and components of Decision Support Systems (DSS).
4. Understand the principles, lifecycle, and architecture of Business Intelligence (BI).
5. Apply concepts of data preparation, integration, and quality management.
6. Explore BI reporting, dashboards, visualization, and emerging BI trends.

Course Outcomes: After successful completion of the course, students will be able to:
1. Identify and classify decision types and environments.
2. Apply decision-making techniques such as what-if, sensitivity, and goal-seek analysis.
3. Explain BI architecture, lifecycle, and workflow.
4. Perform data preparation activities.
5. Design basic BI reports, dashboards, KPIs, and visualizations.
6. Evaluate modern BI trends, cloud BI tools, governance principles, and BI applications

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1		**Foundations of Decision Making**	**07**	
	1.1	Introduction to Decision Making, Types of Decisions: Structured, Semi-structured, Unstructured, Decision-making environments (Certainty, Risk, Uncertainty), Rational & Bounded Rationality Models, Heuristics, Biases & Cognitive Limitations, Role of Information in Organizational Decision Making, Decision Support in Enterprises.		**CO1**
		Self-Learning: Case studies of good vs. poor decisions, Decision frameworks used in industry		
2		**Decision Support Systems (DSS)**	**06**	
	2.1	DSS Concept, Need & Applications, DSS Architecture (database, model base, UI), Types: Data-driven, Model-driven, Knowledge-driven DSS, What-if, Goal-seek, Sensitivity Analysis, Group Decision Support Systems (GDSS), Executive Information Systems (EIS)		**CO2**
		Self-Learning: GDSS in corporate problem solving, DSS tools used in supply chain & healthcare		
3		**Introduction to Business Intelligence**	**06**	
	3.1	What is BI, Evolution & Scope, BI in Strategic, Tactical & Operational Decisions, BI Lifecycle, BI Architecture, BI Workflow: Data → Insights → Action, Role of BI Analysts.		**CO3**
		Self-Learning: Case studies of BI adoption		
4		**Data Preparation & Integration for BI**	**08**	
	4.1	Importance of Data for BI, Data Sources: Enterprise systems, logs, files, cloud sources, ETL Concepts (high level): Extraction techniques, Data Cleaning & Transformation, Loading into BI-ready storage, Data Quality Dimensions (accuracy, completeness, consistency, timeliness), Data Profiling, Metadata Basics (technical vs. business metadata), Master Data & Reference Data Concepts.		**CO4**
		Self-Learning: ETL tools overview (Talend, SSIS, Informatica)		
5		**BI Reporting, Dashboards & Visualization**	**07**	
	5.1	Principles of Data Visualization, BI Reporting Types: Operational, Tactical, Strategic, KPIs, Metrics & Scorecards, Dashboard Design Principles, Real-time & Self-service Reporting Concepts, BI Reporting Tools Overview.		**CO5**
		Self-Learning: Best practices in dashboard layouts, Industry examples of KPI dashboards		
6		**BI Applications & Emerging Trends**	**05**	
	6.1	Cloud BI(overview), BI Service, BI in Domains: Finance, Healthcare, Retail, Manufacturing, Data Governance & Data Stewardship, BI Ethics, Privacy, Responsible Data Use, Performance Management & Balanced Scorecard.		**CO6**

		Self-Learning: Big data basic concepts, Digital transformation using BI		
		Total	**39**	

Textbooks:

1. E. Turban, R. Sharda, and D. Delen, Decision Support and Business Intelligence Systems, 10th ed. Pearson, 2014.
2. J. R. Evans, Business Analytics: Methods, Models, and Decisions, 2nd ed. Pearson, 2016.
3. N. Godbole, Business Intelligence, 1st ed. New Delhi, India: Wiley India, 2013.
4. L. T. Moss and S. Atre, Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications, 1st ed. Boston, MA, USA: Addison-Wesley, 2003.

Reference books:

1. T. H. Davenport and J. Harris, Competing on Analytics: The New Science of Winning, 1st ed. Harvard Business Review Press, 2007.
2. C. Vercellis, Business Intelligence: Data Mining and Optimization for Decision Making, 1st ed. Wiley, 2009. (BI sections only)
3. W. L. Winston, Microsoft Excel Data Analysis and Business Modeling, 4th ed. Microsoft Press, 2014.
4. P. Ponniah, Data Preparation for Analytics Using SAS, 1st ed. Wiley, 2017.

Online References:

1. Microsoft Power BI Learning Path: https://learn.microsoft.com/power-bi/
2. Tableau Official Learning Portal: https://www.tableau.com/learn
3. AWS Cloud BI (QuickSight) Documentation: https://docs.aws.amazon.com/quicksight/

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC6033	Sensor Technology	3			3			3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
MDMC6033	Sensor Technology	20	20	60			100

Pre-requisite:
1. FEC 204- Digital Signal Design

Program Outcomes Addressed
1. PO1: Engineering knowledge.
2. PO2: Problem analysis
3. PO11: Lifelong learning.

Course Objectives: The course aims to enable students:
1. To understand the basics of sensors.
2. To categorize various sensors based on physical parameters.
3. To familiarize with MEMS sensors and Actuators
4. To introduce wireless sensing technologies
5. To develop an understanding of signal conditioning using ADC and DAC
6. To provide insight into various sensor applications

Course Outcomes: Upon completion of this course, Students will be able to:
1. Explain the transduction principle of various sensors.
2. Select sensors suitable for the required application
3. Describe the working of MEMS sensors and Actuators
4. Analyze wireless sensing techniques
5. Design the data acquisition system and identify various signal conditioning methods.
6. Create an application using various sensor technologies

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction**	03	
	1.1	Definition: Need and Role of Sensors		CO1
	1.2	Basic characteristics: sensitivity, resolution, accuracy, linearity, drift, range,		

		hysteresis		
	1.3	Criteria to choose a Sensor: Accuracy, Environmental condition, Range Calibration, Resolution, Cost and Repeatability		
	1.4	Classification of Sensors: Active and Passive Sensors, Contact and Non contacted sensors		
		Self-learning Topics: Introduction to Instrumentation Systems: Block diagram (Sensor, Signal Conditioner, Display/Recorder)		
2.0		**Types of Sensors**	**09**	
	2.1	Temperature Sensors: RTD, Thermocouple and Thermistors sensor		**CO2**
	2.2	Proximity Sensors: Inductive (LVDT), Capacitive, Photoelectric and Ultrasonic sensors		
	2.3	Chemical Sensors: Gas, Smoke, Conductivity and pH sensor		
	2.4	Other Sensors: Optical, Infrared (IR), Sound, Motion, Pressure, Level, Moisture, Humidity, Laser, Image and GPS sensor		
		Self-learning Topics: LDR, Photo Diode and its applications		
3.0		**MEMS Sensors and Actuators**	**06**	
	3.1	MEMS SENSORS: General design methodology, techniques for sensing, Pressure sensor, Mass Flow sensor, Acceleration sensor, Gyroscopes, Micro machined microphones.		**CO3**
	3.2	MEMS ACTUATORS: Techniques for actuation, Digital Micro mirror Devices, Micro Machined Valves		
	3.3	Smart Sensors: Architecture of Smart Sensor and Applications		
		Self-learning Topics: Chemical sensors, Taguchi gas sensor ,Combustible gas sensor		
4.0		**Wireless Sensing Technologies**	**05**	
	4.1	Bluetooth: Concepts of Pico net, Scatter net, Link types, Network Connection Establishments.		**CO4**
	4.2	ZigBee: Components, Architecture, Network Topologies.		
	4.3	Ultra-Wide Band (UWB), Near Field Communication (NFC) and RFID: technical requirements, components and characteristics		
	4.4	WLAN (Wi-Fi): WLAN Equipment, WLAN topologies, IEEE 802.11 Architecture		
		Self-learning Topics: Wireless Sensor Network		
5.0		**Data Acquisition and Signal Conditioning**	**08**	
	5.1	Fundamentals of Data Acquisition: Analog and Digital data acquisition system with different configurations, Data loggers, Noise and interference		**CO5**
	5.2	Signal Conditioning: Wheatstone Bridge, Flash ADC, R2R DAC		
	5.3	Utilization of Signal conditioning circuits for Temperature, Pressure Optical, Strain gauges, Displacement and piezoelectric Transducers		
		Self-learning Topics: Interfacing and Communication with sensors		
6.0		**Sensor Applications**	**08**	
	6.1	Onboard Automobile sensing system, home appliances sensors, Aerospace		**CO6**

		Sensors, Sensors for Environmental Monitoring, Biomedical Sensing Applications		
	6.2	Radio sensors for industrial applications, Radio Astronomy, Remote Sensing, Ground Penetrating Radars, Underwater sensing, LiDAR		
		Self-learning Topics: Case study on Sensor Applications		
		Total	**39**	

Textbooks:

1. D.V.S. Murthy, "Transducers and Instrumentation", PHI Learning, 2nd Edition, 2013.
2. Patranabis, "Sensor and Transducers Prentice Hall", New Delhi, 2013,3rd edition.
3. Ramon Pallas-Areny, John G. Webster, Sensors and Signal Conditioning, 2012, 3rd edition, John Wiley & Sons, New York.
4. Vijay K. Garg, Wireless Communications and Networking, 2015, 2nd edition, Morgan Kaufmann (Elsevier), USA

Reference books:

1. An Introduction to Microelectromechanical Systems Engineering, Nadim Maluf, Kirt Williams, Artech House, 2004.
2. A.K. Sawhney, "A Course in Electrical and Electronic Measurements and Instrumentation", Dhanpatrai & Co., 2015, 20th edition, Dhanpat Rai & Co, New Delhi.
3. Nathan Ida, "Sensors, Actuators and their Interfaces: A Multidisciplinary Introduction", Second Edition, IET Control, Robotics and Sensors Series 127, 2020
4. Jacob Fraden, "Handbook of Modern Sensors Physics, Designs, and Applications, Fifth Edition, Springer, 2015, Springer, New York.

Online References:

1. NPTEL: https://onlinecourses.nptel.ac.in/noc23_ee95/preview

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDMC6063	Organizational Behaviour and Human Resource Management	03	-	01	03	-	01	04

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
MDMC6063	Organizational Behaviour and Human Resource Management	20	20	60	25	--	125

Pre- requisite:

1. Basic understanding of communication and teamwork in project environments
2. Curiosity to learn about human behavior and roles in organizations

Program Outcomes Addressed

1. PO2: Problem Analysis
2. PO7: Global and Ethical Practices
3. PO8: Collaborative Teamwork
4. PO9: Communication
5. PO10: Project Management and Finance
6. PO11: Lifelong Learning

Course Objectives: The course aims to enable students:

1. To explain fundamental concepts of organizational behaviour, including perception, motivation, leadership, teamwork, and communication, relevant to technical and professional environments.
2. To help students understand individual and group behaviour and develop the ability to apply behavioural principles for effective collaboration in engineering teams.
3. To enable students to apply HRM concepts such as recruitment, training, performance appraisal, and employee engagement in engineering and project-based organizations.
4. To develop skills in conflict resolution, negotiation, and decision-making, preparing students to handle workplace challenges in technical settings.
5. To build the ability to analyze organizational culture and structure, and understand how these factors influence innovation, productivity, and technology-driven work environments.
6. To prepare students to become effective engineering professionals by cultivating leadership, ethical behaviour, interpersonal skills, and a people-oriented mindset essential for managing diverse teams.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Relate the key concepts of organizational behaviour and human resource management relevant to engineering workplaces.
2. Understand motivation, communication, and leadership theories to improve individual and team performance in technical environments.
3. Apply workplace situations, identify behavioural issues, and evaluate their impact on engineering teams and project outcomes.
4. Analyse HR practices such as recruitment, performance appraisal, and training, and justify their relevance in engineering industries.
5. Evaluate solutions or strategies for conflict resolution, teamwork enhancement, and employee engagement in engineering projects.
6. Create and demonstrate ethical behaviour, interpersonal skills, and professional conduct, and evaluate their role in shaping an effective organizational culture.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Organizational Behaviour and Interpersonal skills**	**5**	
	1.1	Meaning and significance of OB, functions, roles and skills related to management; contributing disciplines to the field of OB; challenges and opportunities for OB. Importance of interpersonal skills for engineers; verbal and non-verbal communication; active listening; giving and receiving feedback; building trust and rapport; handling difficult conversations		**CO1, CO2**
		Self-learning Topics: Contributing disciplines to OB, Challenges and opportunities in OB, Importance of interpersonal skills for engineers		
2.0		**Personality, Perception and Decision Making**	**6**	
	2.1	Meaning and determinants of personality; major personality traits and attributes; Big Five Model; Job Fit Theory; measuring personality; managing emotions at the workplace. Meaning of perception; factors influencing perception and attitude; Attribution Theory; errors in attribution; link between perception and individual decision making.		**CO2**
		Self-learning Topics: Meaning of perception; factors influencing perception and attitude		
3.0		**Values, Attitudes, Job Satisfaction, Learning and Motivation**	**6**	
	3.1	Definition and types of values; definition and types of attitudes; definition and measurement of job satisfaction; meaning of learning; methods of learning at work. Definition of motivation; theories of motivation (Theory X & Theory Y, Maslow's Need Hierarchy, Two-factor theory, ERG theory, Goal-setting theory, Reinforcement theory, Equity theory, Expectancy theory); motivation from theory to application; link between skill-based plans and motivation theories.		**CO2**
		Self-learning Topics: Definition and types of attitudes; definition and measurement of job satisfaction; meaning of learning; methods of learning at work.		
4.0		**Leadership Theories, Styles & Team Influence, Applied Leadership Skills for Engineering Teams**	**6**	

	4.1	Concept of leadership; leader vs manager; leadership traits and competencies; overview of major leadership theories (trait, behavioural, contingency/situational); leadership styles and their relevance in technical teams. Leading teams; delegation and empowerment; decision-making and problem-solving as a leader; power and influence; ethical leadership; developing one's own leadership plan. **Self-learning Topics:** Differences between a leader and a manager.		**CO2, CO3**
5.0		**Understanding Group Behavior and Teams, Conflicts in the Organization- Organizational Culture and Stress and Organizational Change**	**10**	
	5.1	Definition of groups and teams; Stages of group and team development; Group structure: roles, norms, status; Group tasks and decision making; Types of teams - problem-solving, self-managed, cross-functional, virtual etc.; Features of effective teams; Creating and managing effective teams in engineering projects. Definition of conflict; transitions in conflict thought; functional and dysfunctional conflict; the conflict process; negotiation process; bargaining strategies (distributive & integrative). Organizational Culture and Stress – Definition of organizational culture; types of culture; functions of culture; culture as an asset and liability; creating and sustaining culture; learning organizational culture; Hofstede's model of culture; Organizational Stress – definition of stress; understanding stress and its consequences; sources of stress; managing stress. Meaning of change; forces of change; levels of change; resistance to change; process of change; role of managers in facilitating change. **Self-learning Topics:** Group structure: roles, norms, status; Group tasks and decision making.		**CO4, CO5**
6.0		**Introduction to Human Resource Management & Talent Acquisition Performance Management, Development & Employee Engagement**	**6**	
	6.1	Meaning, scope and objectives of HRM; role of HR in organizations and specifically in engineering / project-based firms; HR planning; job analysis and job descriptions; recruitment and selection basics. Training and development; performance management and feedback; basics of compensation and rewards; employee engagement and welfare; HR's role in supporting career development of engineers. **Self-learning Topics:** Meaning, scope and objectives of HRM, ; HR's role in supporting career development of engineers.		**CO4, CO6**
		Tutorial: Case studies, role play and quiz on organizational behavior and HRM		
		Total	**39**	

Textbooks:

1. Robbins, S. P., & Judge, T. A. (2022). Organizational behavior (19th ed.). Pearson.
2. Aswathappa, K. (2018). Organizational behaviour (12th ed.). McGraw Hill Education.
3. Dessler, G. (2020). Human resource management (16th ed.). Pearson.
4. Armstrong, M. (2021). Armstrong's handbook of human resource management practice (16th ed.). Kogan Page.

Reference books:

1. Luthans, F. (2021). Organizational behavior: An evidence-based approach (14th ed.). McGraw Hill.
2. Schermerhorn, J. R., Hunt, J. G., & Osborn, R. N. (2020). Organizational behavior (14th ed.). Wiley.
3. Mondy, R. W., & Martocchio, J. J. (2016). Human resource management (14th ed.). Pearson.
4. Mathis, R. L., Jackson, J. H., & Valentine, S. R. (2019). Human resource management (15th ed.). Cengage Learning.

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

CIAP (25-Marks):

- At least 06 Tutorials covering entire syllabus must be given during the "Class Wise Tutorial".

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC6011	Machine Vision	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC6011	Machine Vision	20	20	60	--	--	100

Pre- requisite:
1. CEL304: Skill Lab - Python Programming

Program Outcomes Addressed
1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design /Development of Solutions
4. PO4: Conduct Investigations
5. PO6: The Engineer and the world
6. PO11: Life Long Learning.

Course Objectives: The course aims to enable students:
1. To understand the fundamentals of machine vision, including human and computer vision systems, imaging principles, and camera calibration.
2. To analyze and process digital images using enhancement, filtering, and restoration techniques.
3. To extract and model features using edge, corner, and local feature descriptors, and perform feature tracking.
4. To apply image segmentation and object detection techniques and evaluate performance using standard metrics.
5. To implement machine learning and deep learning approaches for image analysis, object detection, and vision-based tasks.
6. To explore emerging trends in industrial vision applications, Industry 4.0 integration, and advanced vision system design.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Analyze computer and human vision systems, image formation models, and camera components, and apply geometric transformations for image representation and manipulation.
2. Apply and Evaluate digital image processing techniques for enhancing and improving visual data quality.
3. Design and implement feature extraction techniques using detectors and descriptors, and analyze their performance in feature matching and tracking algorithms.
4. Evaluate and optimize object segmentation and detection methods using thresholding and region-based approaches for solving real-world problems.
5. Develop, implement, and assess machine learning and deep learning models for image classification, object

detection, and recognition tasks.

6. Create and integrate machine vision solutions using emerging technologies such as Industry 4.0 and 3D imaging, and apply them in practical domains like face recognition, gesture analysis, and medical imaging.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Machine Vision**	**5**	
	1.1	Computer and Human Vision Systems., The Human Eye, Computer versus Human Vision Systems, Evolution of Computer Vision,		**CO1**
	1.2	Image formation, optics and illuminations, Camera: CCD/CMOS, lenses, Camera calibration and geometric transformation.		
		Self-learning Topics: Comparative study of biological vs. artificial vision.		
2.0		**Digital Image Fundamentals**	**6**	
	2.1	Digital Image, Monochrome and Color Images, Image Brightness and Contrast., 2D, 3D, and 4D Images, Digital Image Representation.		**CO2**
	2.2	Image enhancement: histogram equalization, contrast stretching, Spatial domain and frequency domain filtering, Edge detection: Sobel, Prewitt, Canny, Noise models and image restoration basics.		
		Self-learning Topics: Digital Image basics, Monochrome and Color Images, Digital Image File Format.		
3.0		**Feature Extraction and modelling**	**6**	
	3.1	Edges - Canny, LOG, DOG, Line detectors (Hough Transform), Corner detection: Harris, FAST, Local feature descriptors: SIFT, SURF, ORB descriptors, HOG features.		**CO3**
	3.2	Template matching, Feature matching and tracking : KLT(Kanade–Lucas–Tomasi) Tracker, optical flow.		
		Self-learning Topics: Brute-force vs. FLANN-based feature matching		
4.0		**Object Segmentation and detection**	**8**	
	4.1	Difference between classification, detection, and segmentation, Role of segmentation in machine vision pipelines, Challenges: occlusion, lighting variation, scale, cluttered backgrounds, Evaluation metrics: IoU, Precision–Recall, F1-score, mAP (mean Average Precision).		**CO4**
	4.2	Classical Image Segmentation Techniques: Thresholding, Region-Based Segmentation, Edge-Based Segmentation, Morphological Segmentation, Watershed Segmentation.		
		Self-learning Topics: Techniques to improve robustness under challenging conditions.		
5.0	**5**	**Machine Learning & Deep Learning for Vision**	**7**	
	5.1	Introduction to Machine Learning for Vision, Linear Regression, Basic Concepts of Decision Functions, Elementary Statistical Decision Theory,		**CO5**

		Parameter Estimation, Clustering for Knowledge Representation, Dimension Reduction, Linear Discriminant Analysis.		
	5.2	Introduction to Deep Learning & Neural Networks, Convolutional Neural Networks (CNNs), Object Detection Using Deep Learning: R-CNN, YOLO.		
		Self-learning Topics: Semantic segmentation using deep learning (FCN, U-Net), Instance segmentation (Mask R-CNN).		
6.0	**6**	**Emerging Trends in Machine Vision**	**7**	
	6.1	History of Industrial Revolution(s), Machine Vision and Industry 4.0, Emerging Vision Trends in Manufacturing, 3D Imaging, Emerging Vision Trends in Manufacturing		**CO6**
	6.2	Applications in Machine/ Computer Vision: Face detection, face recognition, Gesture recognition, Medical imaging, Vision system design workflow		
		Self-learning Topics: Smart vision sensors and edge computing in factories, Multi-modal imaging (thermal + RGB + depth).		
		Total	**39**	

Textbooks:

1. R. Szeliski, Computer Vision: Algorithms and Applications, 2nd ed., Springer, 2023.
2. D. A. Forsyth and J. Ponce, Computer Vision: A Modern Approach, 3rd ed., Pearson, 2022.
3. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2023.
4. R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed., Pearson, 2023.

Reference books:

1. B. K. P. Horn, Robot Vision, MIT Press, 2020.
2. R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 3rd ed., Cambridge University Press, 2020.
3. S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 5th ed., Pearson, 2022.

Online References:

1. Nptel course on Computer Vision and Image Processing: Computer Vision and Image Processing – Fundamentals and Applications - Course
2. TensorFlow Tutorials (Computer Vision): https://www.tensorflow.org/tutorials/images
3. PyTorch Tutorials (Vision): https://pytorch.org/tutorials/beginner/vision.html
4. IEEE Xplore Digital Library – Computer Vision and Robotics Papers: https://ieeexplore.ieee.org/
5. Towards Data Science – Machine Vision Articles: https://towardsdatascience.com/tagged/computer-vision.

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:
$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC6012	Robotics and Applications	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC6012	Robotics and Applications	20	20	60	--	--	100

Pre- requisite:
1. FEC101: Applied Mathematics - I
2. FEC104: C Programming.

Program Outcomes Addressed
1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO5: Engineering tool Usage
5. PO9: Communication
6. PO11: Life-long Learning.

Course Objectives: The course aims to enable students:
1. To introduce the fundamental concepts of robotics.
2. To develop understanding of robot kinematics.
3. To provide knowledge of robot dynamics and actuation systems.
4. To familiarize learners with robotic sensors and perception.
5. To build competence in robot control, motion planning, and mobile robotics.
6. To develop hands-on understanding of robot programming.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Explain the fundamentals of robotics, robot classification, anatomy, and diverse application areas across industries.
2. Apply kinematic modeling techniques including transformation matrices, forward and inverse kinematics, and DH parameters for robotic manipulators.
3. Analyze robot dynamics and actuation mechanisms using principles of force, torque, Newton–Euler, and Lagrangian methods and evaluate different actuator technologies.
4. Evaluate and integrate various robotic sensors and perception systems, including vision processing and basic sensor fusion concepts.
5. Design and analyze robot control strategies and motion planning algorithms and interpret mobile robot kinematics and navigation concepts.

6. Develop and implement basic robot programming using ROS, simulation tools, and AI/ML techniques for robotic applications.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to Robotics**	**06**	
	1.1	Definition, scope, importance of robotics, History and evolution of robotic systems		**CO1**
	1.2	Classification of robots: industrial, service, mobile, humanoid, Anatomy of a robot: links, joints, end-effectors, DOF, workspace		
	1.3	Industrial applications: manufacturing, healthcare, agriculture, defense, logistics, Overview of automation & Industry 4.0		
		Self-learning Topics: Soft robotics: materials, applications, emerging trends		
2.0		**Robot Kinematics**	**07**	
	2.1	Coordinate frames and transformation matrices		**CO2**
	2.2	Forward kinematics, Inverse kinematics (analytical concepts)		
	2.3	Denavit–Hartenberg (D–H) parameters,Homogeneous transformations, Kinematic redundancy, singularities		
		Self-learning Topics: Jacobian matrix: derivation, geometric vs analytical Jacobian, Velocity kinematics & differential kinematics		
3.0		**Robot Dynamics & Actuation**	**06**	
	3.1	Concepts of force, torque, inertia, Dynamic modeling of manipulators		**CO3**
	3.2	Newton–Euler and Lagrangian formulations (conceptual)		
	3.3	Actuation systems**:** Electric- DC motors, servo motors, stepper motors Pneumatic & Hydrolic Systems.		
	3.4	Power transmission: gears, belts, harmonic drives.		
		Self-learning Topics: Dynamic simulation using Gazebo.		
4.0		**Sensors, Perception & Vision Systems**	**07**	
	4.1	Sensors: Proximity, IR, ultrasonic, tactile, force/torque sensors		**CO4**
	4.2	Encoders, IMU, gyroscope, LIDAR and Radar basics.		
	4.3	Perception & Vision: Image acquisition and preprocessing, Feature extraction, segmentation.		
	4.4	Object detection and recognition, Basics of sensors, Sensor types and working.		
		Self-learning Topics: 3D perception: depth cameras (Intel RealSense), stereo vision		
5.0		**Control system & Mobile Robotics**	**07**	
	5.1	Introduction to control systems, Mathematical modelling basics, Standard test signals, Time domain response characteristics, frequency domain basics.		**CO5**
	5.2	Robot Control: Open-loop and closed-loop control, PID controller: concept and tuning, Joint space vs task space control.		

	5.3	Mobile Robotics: Wheeled robot models – differential drive, skid steer, Kinematics of mobile robots, Basics of SLAM (mapping & localization), Autonomous navigation architecture.		
		Self-learning Topics: Motion Planning: Trajectory generation, Path planning algorithms: BFS, DFS, A*, Dijkstra, RRT (conceptual).		
6.0		**Robot Programming, AI in Robotics & Applications**	**06**	
	6.1	Robot Programming: Basics of Robot Operating System (ROS), ROS nodes, topics, publishers/subscribers.		**CO6**
	6.2	Simulation tools: Gazebo, Webots, CoppeliaSim, Programming robots using Python/C++.		
	6.3	AI & Applications: Role of AI & ML in robotics, Reinforcement learning for robot decision-making.		
	6.4	Collaborative robots (CoBots), Use cases: smart factories, autonomous vehicles, agriculture, logistics, defense.		
		Self-learning Topics: Ethics in robotics & AI, I-based motion planning (Neural Motion Planner basics)		
		Total	**39**	

Textbooks:

1. S. K. Saha, Introduction to Robotics, McGraw Hill Education.
2. Spong, Hutchinson & Vidyasagar, Robot Modeling and Control, Wiley.
3. Robert Shilling, “Fundamentals of Robotics - Analysis and control, Prentice Hall of India, 2009.
4. Saeed Benjamin Niku, “Introduction to Robotics – Analysis, Control, Applications”, Wiley India Pvt. Ltd., Second Edition, 2011.

Reference books:

1. John J. Craig, “Introduction to Robotics – Mechanics & Control”, Third Edition, Pearson Education, India,2009.
2. Mark W. Spong , Seth Hutchinson, M. Vidyasagar, “Robot Modeling & Control ”, Wiley India Pvt. Ltd., 2006.
3. Mikell P. Groover et.al, ”Industrial Robots-Technology, Programming & applications”, McGraw Hill , New York, 2008.
4. S. R. Deb and Sankha Deb, “Robotics Technology and Flexible Automation” , Second Edition. TMH.

Online References:

1. ROS Tutorials – Official Documentation https://wiki.ros.org/ROS/Tutorials.
2. NPTEL Courses: Robotics by IIT Kanpur, Robot Kinematics and Dynamics by IIT Kharagpur.
3. Coursera / edX (Free audit mode): Robotics Specialization (University of Pennsylvania), Modern Robotics (Northwestern University).
4. MATLAB Robotics Toolbox (Peter Corke) https://petercorke.com/toolboxes/robotics-toolbox/
5. Gazebo & Webots Official Documentation. https://gazebosim.org, https://cyberbotics.com

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC6013	Digital Forensics	03	-	-	03	-	-	03

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC6013	Digital Forensics	20	20	60	-	-	100

Pre- requisite:
1. CEC503: Computer Network
2. CEPEC5013: Ethical Hacking

Program Outcomes Addressed
1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Conduct Investigations of Complex Problems
4. PO4: Engineering Tool Usage
5. PO5: The Engineer and The World
6. PO6: Individual and Collaborative Team work

Course Objectives: The course aims to enable students:
1. Basic understanding of digital forensics and incident response.
2. Knowledge of digital evidence, admissibility, and acquisition methods.
3. Skills to analyze hard disks, memory, routers, and malware.
4. Ability to investigate Windows and Unix system artifacts.
5. Competence in mobile device, SIM, and GPS forensics.
6. Understanding of social media forensic investigation.

Course Outcomes: Upon completion of this course, Students will be able to:
1. Describe digital forensics and incident response fundamentals.
2. Apply appropriate techniques to identify, acquire, and duplicate digital evidence from various platforms.
3. Analyze storage, memory, network, and malware evidence.
4. Use system artifacts and logs to conduct forensic examination of Windows and Unix systems.
5. Apply mobile forensic methods for device, SIM, and GPS data.
6. Collect and analyze evidence from social media platforms to support investigations.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Digital Forensics Fundamentals and Incident Response**	**6**	
	1.1	Introduction to Digital Forensics and its Objectives, Understanding Digital Evidence Types and Rules, Incident Response Methodology and The Role of CSIRT.		**CO1**
	1.2	Introduction to Digital Forensics, Digital Evidence Types, Incident Response Methodology, CSIRT Roles, Pre-Incident Preparation and Incident Response Process.		
		Self-learning Topics: Open-source IR tools: GRR Rapid Response, OSQuery.		
2.0		**Digital Evidence, Forensics Duplication and Digital Evidence Acquisition**	**9**	
	2.1	Digital Evidence, Types of Digital Evidence, Challenges in Acquiring Digital Evidence, Admissibility of Evidence, Challenges in Evidence Handling, Chain of Custody.		**CO2**
	2.2	Digital Forensics Examination Process - Seizure, Acquisition, Analysis, Reporting. Necessity of Forensic Duplication, Forensic Image Formats, Forensic Duplication Techniques.		
	2.3	Data Acquisition: Introduction to Static and Live/Volatile Data, Static Data Acquisition from Windows (FTK Imager), Static Data Acquisition from Linux (Dd/Dcfldd), Live Data Acquisition from Windows (FTK Imager), Network Forensics (Wireshark).		
		Self-learning Topics: File System Basics (NTFS, EXT4, APFS), Network Forensics Tools.		
3.0		**Forensics Investigation and Analysis**	**4**	
	3.1	Analyzing Hard Drive Forensic Images, Analyzing RAM Forensic Image, Investigating Routers.		**CO3**
	3.2	Malware Analysis - Malware, Viruses, Worms, Essential Skills and Tools for Malware Analysis, List of Malware Analysis Tools and Techniques, Ransomware analysis, Case Study: SimpleLocker Ransomware Analysis.		
		Self-Learning Topics: Introduction to Malware Sandboxes: Cuckoo Sandbox, Malware Behavior Classification.		
4.0		**Windows and Unix Forensics Investigation**	**8**	
	4.1	Investigating Windows Systems - File Recovery, Windows Recycle Bin Forensics, Data Carving, Windows Registry Analysis, USB Device Forensics, File Format Identification, Windows Features Forensics Analysis, Cortana Forensics.		**CO4**
	4.2	Investigating Unix Systems - Reviewing Pertinent Logs, Performing Keyword Searches, Reviewing Relevant Files, Identifying Unauthorized User Accounts or Groups, Identifying Rogue Processes, Checking for Unauthorized Access		

		Points.		
		Self-learning Topics: Timeline Analysis, Unix Tool: Logwatch		
5.0		**Mobile Forensics**	**8**	
	5.1	Android Forensics, Mobile Device Forensic Investigation - Storage Location, Acquisition Methods, Data Analysis.		**CO5**
	5.2	GPS Forensics - GPS Evidentiary Data, GPS Exchange Format (GPX), GPX Files, Extraction of Waypoints and Trackpoints, Display Tracks on a Map.		
	5.3	SIM Cards Forensics - The Subscriber Identification Module (SIM), SIM Architecture, Security, Evidence Extraction.		
		Self-Learning Topics: Android File System Structure, Pysim (Tool for SIM Forensics).		
6.0		**Social Network Forensics and Evidence Collection**	**4**	
	6.1	Facebook & Twitter/X: Overview, Finding People, Obtaining Data, Privacy Levels and Access, Case Study.		**CO6**
	6.2	LinkedIn/Pinterest: Introduction, User Demographics, Finding People, Obtaining Data, Privacy Levels and Access, Case Study.		
		Self-Learning Topics: OSINT Framework, Understanding Metadata in Social Media Posts.		
		Total	**39**	

Textbooks:

1. J. T. Luttgens, M. Pepe, and K. Mandia, Incident Response & Computer Forensics, 3rd ed. New York, NY, USA: McGraw-Hill Professional, 2014.
2. N. A. Hassan, Digital Forensics Basics: A Practical Guide Using Windows OS. Berkeley, CA, USA: Apress, 2019.
3. X. Lin, Introductory Computer Forensics: A Hands-on Practical Approach. Cham, Switzerland: Springer Nature, 2018.
4. J. Golbeck, Introduction to Social Media Investigation: A Hands-on Approach. Syngress, 2015.

Reference books:

1. M. Ligh, A. Case, J. Levy, and A. Walters, The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory. Hoboken, NJ, USA: Wiley, 2014.
2. T. S. Ho and S. Li, Eds., Digital Forensics of Multimedia Data and Devices. Hoboken, NJ, USA: Wiley, 2015.
3. J. R. Vacca, Ed., Computer and Information Security Handbook, 3rd ed. Amsterdam, Netherlands: Elsevier, 2017.
4. S. Bommisetty, R. Tamma, and H. Mahalik, Practical Mobile Forensics, 4th ed. Birmingham, U.K.: Packt Publishing, 2023.
5. N. Jain and D. Kalbande, Digital Forensics. Hoboken, NJ, USA: Wiley, 2023.

Online References:

1. Coursera: EC-Council Digital Forensics Essentials: https://www.coursera.org/learn/digital-forensics-essentials-dfe?
2. Coursera: Digital Forensics Course Collection: https://www.coursera.org/courses?query=digital%20forensics
3. SWAYAM: Information Security and Cyber Forensics Course: https://onlinecourses.swayam2.ac.in/cec21_ge10/preview?
4. Great Learning – Cyber Forensics Course: https://www.mygreatlearning.com/academy/learn-for-free/courses/cyber-forensics?

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEC6014	Natural Language Processing	3	--	--	3	--	--	3

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE$			
		ISE	MSE				
CEPEC6014	Natural Language Processing	20	20	60	--	--	100

Pre- requisite:

1. CEC501: Theoretical Computer Science

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct Investigations of Complex Problems
5. PO11: Life-Long Learning

Course Objectives: The course aims to enable students:

1. To introduce key concepts, linguistic foundations, text pre-processing and core challenges in Natural Language Processing
2. To build student competency in morphological analysis and language modeling.
3. To explain syntactic structures, tagging and parsing methods for linguistic analysis.
4. To introduce key semantic and discourse concepts for understanding word meaning and document structure.
5. To explore the essential architectures and training approaches underlying large language models.
6. To examine key NLP applications such as NER, QA, machine translation, and modern large-scale language models.

Course Outcomes: Upon completion of this course, Students will be able to:

1. Apply foundational NLP concepts and basic text-processing techniques to analyze language data and develop simple NLP applications.
2. Apply finite-state techniques and N-gram models to analyze morphology and evaluate language models effectively.
3. Analyze POS tagging methods and parsing techniques to interpret and construct accurate syntactic structures of natural language text.
4. Interpret semantic and discourse techniques to understand text meaning and resolve references.
5. Describe key LLM architectures and evaluation methods used in modern NLP systems.
6. Identify major NLP applications and modern LLMs used for real-world language understanding tasks.

Module No.	Unit No.	Topics	Hrs.	Mapped to Course Outcome
1.0		**Introduction to NLP and Text Processing**	**06**	
	1.1	What is Natural Language Processing (NLP)? History of NLP, Generic NLP System, Levels of NLP, Ambiguity and Layers of NLP. Challenges of NLP, Processing Indian Languages, Applications of NLP.		**CO1**
	1.2	Basic Text Processing: Tokenization, Stopword Removal, Stemming and Lemmatization, Regular Expressions		
		Self-learning Topics: Tools for regional languages pre-processing and other functionalities.		
2.0		**Word Level Analysis**	**06**	
	2.1	Survey of English Morphology, Inflectional Morphology, Derivational Morphology.		**CO2**
	2.2	Morphological Models: Dictionary lookup, finite state morphology; Morphological parsing with FST (Finite State Transducer); Lexicon free FST Porter Stemmer algorithm		
	2.3	Language models- N-gram models and smoothing techniques: Laplace smoothing, Good-Turing Discounting, Evaluating language models- perplexity, cross-entropy		
		Self-learning Topics: Noisy channel models, Edit distances, Advance Issues in Language Modelling		
3.0		**Syntax Analysis**	**10**	
	3.1	Part-of-speech Tagging: Illustration of Ambiguity in POS, Table Look-up-based and Rule-based POS, Statistical POS Tagging: Hidden Markov Models (HMM) for POS tagging, Viterbi decoding algorithm; Chunking		**CO3**
	3.2	Context-Free Grammars (CFG) and parse trees; Linguistics of Parsing		
	3.3	Algorithmics of Parsing, Constituency Parsing: Rule Based (Top-Down, Bottom-Up, CYK)		
	3.4	Statistical Parsing, Dependency Parsing and Neural Parsing		
		Self-learning Topics: Evaluating parsers, Parsers based language modelling, regional languages POS tree banks		
4.0		**Semantic and Discourse Processing**	**08**	
	4.1	Introduction, Meaning Representation; Lexical Semantics; Study of Various language dictionaries like WordNet, Lexical Relations		**CO4**
	4.2	Word Embeddings: Word2Vec, CBOW, Skip-gram, GloVe		
	4.3	Word Sense Disambiguation (WSD) methods: Lesk algorithm Document representation: TF-IDF, Doc2Vec		
	4.4	Discourse Structure, Cohesion, Discourse Coherence & Structure , Reference resolution, Reference phenomena, Anaphora resolution using Hobbs algorithm		

		Self-learning Topics: Dictionaries for regional languages, Distributional Semantics, Topic Models , Multimodal Discourse (Text + Image, Text + Video)		
5.0		**Modern NLP with Transformers**	**05**	
	5.1	Introduction to Large Language Models (LLM), Fundamental components of the Transformer model		**CO5**
	5.2	Transformer Architecture and Self-Attention Mechanism, Encoder–Decoder architecture, Introduction to pre-training and fine-tuning paradigms, Pretrained models (BERT, GPT, T5) and fine-tuning, Positional Encoding. Evaluation metrics (BLEU, ROUGE, F1)		
		Self-learning Topics: Transformer Variants-ALBERT, XLNet, Long former.		
6.0		**Applications of NLP**	**04**	
	6.1	Named Entity Recognition (NER) , Question Answering System, Machine Translation		**CO6**
	6.2	Advanced NLP applications: GPT- 4, Llama-3, Claude-3, Mistral and Gemini.		
		Self-learning Topics: Use of NLP in Recommender Systems, Federated Learning for NLP Applications, Develop Chatbot Applications.		
		Total	**39**	

Textbooks:

1. Daniel Jurafsky, James H. and Martin, Speech and Language Processing, Third Edition, Prentice Hall, 2026.
2. Christopher D.Manning and HinrichSchutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.
3. Pushpak Bhattacharyya and Aditya Joshi, Natural Language Processing, Wiley India Pvt. Ltd., Edition: 2023
4. Dr. Tanmoy Chakraborty, “Introduction to Large Language Models: Generative AI for Text” , Wiley (Wiley India). December 2024.

Reference books:

1. Siddiqui and Tiwary U.S., Natural Language Processing and Information Retrieval, Oxford University Press, 2008.
2. Daniel M Bikel and ImedZitouni — Multilingual natural language processing applications: from theory to practice, IBM Press, 2013.
3. Palash Goyal, Sumit Pandey, Karan Jain. Deep Learning for Natural Language Processing-2018
4. Joseph Babcock, Raghav Bali, “Generative AI with Python and TensorFlow”, Packt Publishing-2021.

Online References:

1. NPTEL Course”Natural Language Processing”: https://onlinecourses.nptel.ac.in/noc23_cs45/preview
2. NPTEL Swayam “Natural Language Processing” : https://archive.nptel.ac.in/courses/106/106/106106211/
3. Natural Language Toolkit Book: https://www.nltk.org/book/
4. Stanford Course”Natural Language Processing with Deep Learning”: https://web.stanford.edu/class/cs224n/

Course Assessment:

ISE:

- To be conducted in any of these forms - Assignment/ Quiz/ Presentation/ Class Test/ Case study etc. of 20 marks.
- ISE 20 marks = 05 marks for attendance + 15 marks for activities.

MSE:

- To be conducted as written examination for 20 marks (on 40% - 50% syllabus)

End Semester Examination:

$ ESE of duration 03 hours are of 80 marks and scaled to 60.

1. Question paper will comprise of 03 questions.
2. Question1(20 marks): - Solve any 04 out of 06. All questions carry 05 marks each.
3. Question 2 (40 marks): - Solve any 04 out of 06. All questions carry 10 marks each.
4. Question3(20 marks):- Solve any 04 out of 06. All questions carry 05 marks each.
5. All COs should be mapped as per the weightage in the syllabus.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEL601	Computational Intelligence Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEL601	Computational Intelligence Lab	--	--	--	25	25	50

Pre-requisite:
1. CEL502: Computer Network Lab
2. FEL103: C-Programming Lab
3. CEL304: Skill Lab – Python Programming

Program Outcomes Addressed:
1. PO1: Engineering knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct investigations of complex problems
5. PO5: Engineering tool usage
6. PO7: Ethics
7. PO8: Individual and Collaborative Team work
8. PO11: Lifelong learning

Lab Objectives: The course aims to enable students:
1. Gain hands-on experience with cryptographic and network security mechanisms.
2. Practice secure communication, authentication, and access control techniques.
3. Explore network attacks and security tools in a controlled environment.
4. Work with AI problem-solving, reasoning, and inference methods.
5. Develop intelligent solutions using neural networks and fuzzy systems.
6. Integrate security, AI, and soft computing concepts through laboratory experiments.

Lab Outcomes: Upon completion of this course, Students will be able to:
1. Implement cryptographic algorithms and secure communication techniques.
2. Apply key exchange, digital signatures, and encryption modes effectively.
3. Simulate and analyze network security attacks and their impact on systems and networks.
4. Demonstrate role-based access control through user, role, and permission management.
5. Solve problem statements using search, inference, and probabilistic techniques.
6. Design and implement neural network and fuzzy logic–based intelligent systems.

Suggested List of Experiments: Students are required to complete at least 12 experiments.		
Sr. No.	**Title of Experiments**	**LO Mapped**
1	A communication system requires a secure method to protect sensitive data from unauthorized access. The goal is to design and evaluate a classical encryption mechanism that enhances security by combining multiple cryptographic techniques to resist basic cryptanalysis attacks. **Objectives:** a) To develop a combined encryption scheme by integrating substitution and transposition techniques with appropriate key selection. b) To apply the designed cipher on sample inputs and verify correctness of both encryption and decryption processes. c) To examine resistance against frequency analysis and pattern-based attacks.	LO1
2	An organization requires a secure encryption mechanism to protect large volumes of sensitive data during transmission. The objective is to implement symmetric encryption techniques and evaluate how different modes of operation impact data security and confidentiality. **Objectives:** a) To apply symmetric encryption to secure large data inputs using standard cryptographic techniques. b) To implement and configure multiple operation modes (ECB, CBC, CFB) to process encrypted data.	LO1
3	Two communicating parties need to securely exchange cryptographic keys over an insecure network where adversaries may intercept or manipulate communication. The aim is to implement a secure key exchange mechanism that enables both parties to establish a shared secret without directly transmitting it. **Objectives:** a) To implement a secure key exchange mechanism to generate a shared secret between communicating parties. b) To Simulate secure key generation between sender and receiver.	LO2
4	A network administrator wants to understand the impact of Denial-of-Service attacks on system availability. Students are required to simulate and analyze this attack in a controlled environment to observe their effect on network performance and service accessibility. **Objectives:** a) To simulate a DoS attack using tools such as hping or hping3. b) To analyze network traffic during the attack. c) To observe system performance degradation. d) To recommend mitigation techniques to prevent DoS attacks.	LO3
5	An organization wants to enforce access control policies to restrict unauthorized access to resources. The system requires implementation and analysis of access	LO4

	control mechanisms to ensure that only authorized entities can access specific system resources. **Objectives:** a) To implement Role-Based Access Control (RBAC). b) To create users, roles, and permission sets. c) To assign roles to users and enforce access restrictions. d) To analyze how access is controlled based on roles.	
6	A problem-solving system needs to find optimal solutions using intelligent search techniques in a state-space environment where multiple paths may exist between the initial and goal states. **Objectives:** a) To implement an informed search algorithm (e.g., A*, Greedy Search). b) To apply it to problems such as N-Queen or 15-Puzzle problem, Water Jug Problem, Travelling Salesman problem.	LO5
7	An AI system must solve problems without prior knowledge using systematic exploration techniques in a state-space environment where all possible paths must be explored to reach a solution. **Objectives:** a) To implement an uninformed search algorithm (e.g., BFS, DFS). b) To apply it to problems such as N-Queen or 15-Puzzle problem, Water Jug Problem, Travelling Salesman problem.	LO5
8	An intelligent system needs to derive conclusions from a given knowledge base using logical reasoning to infer new information and support decision-making. **Objectives:** a) To apply forward chaining and backward chaining methods. b) To implement resolution inference in First Order Predicate Logic (FOPL). c) To derive goal statements from given facts. d) To analyze the efficiency of different inference techniques.	LO5
9	An organization aims to automate decision-making using intelligent agents that can perceive the environment, take appropriate actions, and optimize outcomes based on defined objectives. **Objectives:** a) To design and implement an AI agent using open-source tools. b) To define environment, states, and actions. c) To simulate agent behavior for a given problem. d) To evaluate agent performance and decision-making capability.	LO5
10	A decision-making system needs to model uncertainty and probabilistic relationships among variables to support accurate reasoning under uncertain conditions. **Objectives:** a) To create a Bayesian Network for a given problem. b) To define nodes, dependencies, and probabilities.	LO5

	c) To perform inference using the network. d) To analyze decision outcomes under uncertainty.	
11	An engineer aims to design a learning-based system using neural networks to model input-output relationships and perform predictive analysis. The system requires constructing a feedforward neural network architecture and applying an appropriate learning rule for training. **Objectives:** a) To build a feedforward neural network architecture. b) To implement a suitable learning rule. c) To train the network using sample data. d) To evaluate performance based on accuracy and error rate.	LO6
12	A system requires handling uncertainty using fuzzy logic instead of crisp values to represent imprecise and vague information. The system must model such uncertainty using appropriate fuzzy membership functions. **Objectives:** a) To implement fuzzy membership functions. b) To define linguistic variables and rules. c) To analyze the behavior of fuzzy sets. d) To compare fuzzy logic with classical logic.	LO6
13	A control system needs to make decisions based on approximate reasoning under uncertain conditions. The system must utilize fuzzy logic to process imprecise inputs and generate appropriate control actions. **Objectives:** a) To design and implement a fuzzy control system. b) To define input/output variables and rule base. c) To simulate system behavior for different inputs. d) To analyze system performance and accuracy.	LO6
14	A secure communication system requires encryption and authentication using public key cryptography to ensure confidentiality, integrity, and authenticity of transmitted data. The system must implement and analyze cryptographic mechanisms for secure data exchange and digital signing. **Objectives:** a) To implement RSA or ElGamal cryptosystem. b) To perform encryption and decryption operations. c) To implement digital signature generation and verification. d) To analyse security and computational efficiency.	LO2
15	An organization requires secure email communication using encryption and authentication mechanisms to ensure confidentiality, integrity, and authenticity of messages during transmission. The system must implement and analyze secure email exchange using cryptographic tools. **Objectives:** a) To explore and configure GPG tool in Linux.	LO1

	b) To encrypt and decrypt email messages. c) To implement digital signatures for authentication.	

Textbooks:

1. W. Stallings, Cryptography and Network Security: Principles and Practice, 8th ed. Boston, MA, USA: Pearson, 2022. E. Cole, Network Security Bible, 2nd ed. Hoboken, NJ, USA: Wiley, 2011.
2. S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Upper Saddle River, NJ, USA: Pearson Education, 2021.
3. S. Rajasekaran and G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications, 2nd ed. New Delhi, India: PHI Learning, 2017.

Reference books

1. M. Gregg, Build Your Own Security Lab: A Field Guide for Network Testing, 1st ed. Indianapolis, IN, USA: Wiley, 2008.
2. T. Boyles, CCNA Security Study Guide: Exam 640-553, 1st ed. Hoboken, NJ, USA: Wiley/Sybex, 2010.
3. E. Rich and K. Knight, Artificial Intelligence, 5th ed. New Delhi, India: Tata McGraw-Hill Education, 2025.
4. M. T. Hagan, H. B. Demuth, and M. H. Beale, Neural Network Design, 2nd ed. Boston, MA, USA: Cengage Learning, 2014.

Online References:

1. Cryptography & Network Security- https://nptel.ac.in/courses/106105162
2. An Introduction to Artificial Intelligence – https://share.google/umcQLMykKscKcgb5k
3. Introduction to Soft Computing –NPTEL+ https://share.google/FWefkZDqu5BIHkiCn
4. Hugging Face – AI Models & Examples-https://huggingface.co/

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered.)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

- End-semester Practical and oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEL602	Skill Lab-Cloud Computing	--	02*+ 02	--	--	02	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEL602	Skill Lab-Cloud Computing	--	--	--	25	25	50

Pre-requisite:

1. CEC402 - Operating System
2. CEC503 - Computer Network

Program Outcomes:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design and development of solutions
4. PO5: Modern tool usage
5. PO9: Individual and teamwork
6. PO11: Lifelong learning

Lab Objectives: The course aims to enable students:

1. To conceptualize cloud computing fundamentals.
2. To make students familiar with key concepts of virtualization.
3. To familiarize students with different cloud service models, including IaaS, PaaS, and SaaS.
4. To provide students with an understanding of Security as a Service (SECaaS).
5. To introduce students to serverless computing concepts and machine learning workflows.
6. To acquaint students with containerization technologies, orchestration, and application security and infrastructure monitoring in DevSecOps.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Implement different types of virtualization techniques.
2. Analyze various cloud computing service models and implement them to solve the given problems.
3. Identify major security issues in the cloud and provide appropriate security solutions.
4. Implement Serverless architecture, configure workflows, and apply basic machine learning workflows.
5. Implement containerized applications and perform static application security testing integrated with CI/CD pipelines.
6. Configure and analyze monitoring of infrastructure and services to ensure availability and reliability in DevOps systems.

Module No.	Unit No.	Topics	Hrs.	LO Mapped
1		**Introduction to Cloud Computing**	**03**	
	1.1	Definition of Cloud Computing, Need for Cloud computing, Characteristics, NIST model		**LO1**
	1.2	Cloud deployment models (Public, Private, Hybrid, Community) and Service Models: IaaS, PaaS, and SaaS.		
	1.3	Advantages and Disadvantages of Cloud Computing.		
		Self-Learning: Compare cloud vs traditional computing.		
2		**Virtualization & Hypervisors**	**04**	
	2.1	Introduction and benefits of virtualization, Virtualization Structures and Implementation Levels, Virtual Machine Monitor (VMM).		**LO2**
	2.2	Hypervisors —Type I, II (Xen, KVM, VMware ESXi, Hyper-V), Virtualization Mechanisms and Tools, Hosted vs Bare Metal Hypervisors.		
		Self-Learning: Virtualization vs Containerization.		
3		**Amazon Core Services**	**08**	
	3.1	Introduction to the AWS Cloud, Compute service: Introduction to EC2, Instance Types, instance lifecycle, Auto Scaling, Load Balancing.		**LO3**
	3.2	Storage service: Introduction to S3, working with Buckets, setting bucket security, bucket properties, working with Elastic Block Store (EBS) Volumes, and Object Storage Vs Block Storage.		
	3.3	Network as a Service: Introduction to virtual private cloud (VPC), Subnets, Elastic Network Interfaces, Internet Gateways, Route Tables, Security Groups, and Network ACLs.		
	3.4	Database as a Service: Introduction to Amazon Relational Database Service (RDS), Database Engines, Non-relational (No-SQL) Databases, Types of Non-relational Databases, and their features.		
	3.5	Deployment & Management Services: Amazon Elastic Beanstalk		
		Self-Learning: Comparison of AWS services with other cloud service platforms. Selection criteria for cloud vendors.		
4		**Cloud Security & Identity Management**	**04**	
	4.1	Security service: AWS Identity and Access Management (IAM). Amazon Cognito: Provides user authentication, authorization, and user management for web and mobile applications via user pools. AWS Key Management Service (KMS): A managed service for creating, storing, and controlling encryption keys.		**LO4**
	4.2	AAA Administration for Clouds - AAA model, SSO for Clouds, Authentication management, and Authorization management in clouds.		
		Self-Learning: Cloud Security in AWS.		
5		**Containerization & Orchestration**	**03**	
	5.1	Containerization using Docker - Docker containerization concepts, image building, running containers, Docker architecture		**LO5**

	5.2	Kubernetes - Kubernetes orchestration, Kubernetes architecture, steps to deploy a Kubernetes Cluster on local systems, deploy applications on Kubernetes, and create a Service in Kubernetes		
		Self- Learning: Basics of CI/CD pipeline.		
6		**Serverless & ML on Cloud**	**04**	
	6.1	Serverless Computing: Basics of Serverless Computing, Working with Serverless environment, Introduction to serverless events and functions, AWS Lambda. Application services: AWS Simple Notification Service (SNS), Simple Queue Service (SQS), and Simple Email Service (SES).		**LO6**
	6.2	Machine Learning with SageMaker: Introduction to SageMaker Studio, preparing datasets, and training ML models using built-in algorithms.		
		Self-Learning: Explore real-world use cases and case studies on sustainability and green cloud practices.		
		Total	**26**	

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	**Introduction to Virtualization.** To study and implement Hosted Virtualization using VirtualBox& KVM. Objective: • To know the concept of Virtualization along with its types, structures, and mechanisms. • To create and run Virtual machines inside hosted hypervisors like VirtualBox and KVM with their comparison based on various virtualization parameters.	LO1
2	**Introduction to Virtual compute and block storage services.** To deploy and manage compute and block storage services in the cloud. Objectives: • Launch and configure compute instances • Manage security groups and key pairs • Attach, mount, and snapshot EBS volumes	LO2
3	**Build Your Virtual Private Cloud (VPC) and Launch a Web Server.** To design and deploy a custom VPC and launch an EC2 instance. Objectives: • Create VPC and subnets • Configure security group • Launch EC2 instance in VPC	LO2
4	**Introduction to Object Storage Services.** To learn basic features of object storage services. Objectives: • Create a bucket using any cloud platform. • Upload objects. • Manage permissions and bucket policies .	LO2
5	**Build Your DB Server and Interact With Your DB Using an App.**	LO2

	To deploy and configure a relational database instance. Objectives: • Launch database instance • Configure DB connectivity • Interact with DB using web app	
6	**Automating Infrastructure Deployment with AWS CloudFormation.** To automate deployment using CloudFormation. Objectives: • Deploy VPC and app layers • Explore templates in Designer • Delete stack with retention policy	LO2
7	**Cloud Security as a Service (SECaaS):** identity and access management (IAM)**, Application Authentication, and Data Encryption** To implement cloud security mechanisms using identity and access management (IAM) principles and user authentication/authorization services for secure resource access. Objectives: • Create IAM users/groups/roles & test policy effects • Add Cognito user pool for app authentication • Enable encryption for S3/EBS data-at-rest	LO3
8	**Building Decoupled Applications by Using Amazon SQS.** To build a decoupled architecture using SQS and SNS. Objectives: • Configure S3 events to SNS • Subscribe SQS to SNS • Implement message polling	LO4
9	**Implementing a Serverless Computing Architecture.** To understand serverless computing concepts, including function execution workflows, triggers, and deployment of functions using programming languages such as Python, Java, or Node.js. Objectives: • Invoke Lambda from S3/DynamoDB • Configure SNS notifications • Build serverless workflow and create your first Lambda functions using Python / Java / Nodejs.	LO4
10	**To train and develop a Machine Learning Model.** To train, evaluate, and deploy a machine learning model in a cloud-based environment using supervised/unsupervised learning algorithms (e.g., gradient boosting) on a dataset. Objectives: • Split data into training, validation, and test datasets • Train a model using a machine learning algorithm. • Deploy a machine learning model and perform testing.	LO4
11	**To study and Implement Containerization and Orchestration.** To deploy cloud-native applications using containerization and orchestration Objectives: • Build and run Docker containers	LO5

	• Create Kubernetes deployments and services • Scale and manage containerized applications	
12	**Static Application Security Testing (SAST) using Jenkins and SonarQube.** To understand the Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab. Objectives: • Perform static code analysis to identify vulnerabilities, code smells, and security issues. • Integrate Jenkins with SonarQube to automate SAST during the build process.	LO5
13	**Dynamic Application Security Testing (DAST) using OWASP ZAP.** To perform dynamic security testing of web applications using OWASP ZAP. Objectives: • Identify runtime web vulnerabilities • Perform automated and manual security scans • Analyze OWASP Top-10 vulnerabilities	LO5
14	**Creating a Jenkins CI/CD Pipeline with SonarQube for Static Code Analysis (SAST)** Create a Jenkins CI/CD Pipeline with SonarQube / GitLab integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application. Objectives: • Create and configure a Jenkins pipeline for building a sample Web / Java / Python application. • Integrate SonarQube with Jenkins to automate static code analysis (SAST) during the pipeline execution.	LO5
15	**Continuous Monitoring using Nagios Core and Plugins.** To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins, and NRPE (Nagios Remote Plugin Executor) on Linux Machine. Objectives: • Install and configure Nagios Core on a Linux server. • Install and manage Nagios Plugins for system and service monitoring. • Configure NRPE (Nagios Remote Plugin Executor) for remote host monitoring.	LO6
16	**Server and Service Monitoring using Nagios.** To perform Port, Service monitoring, and Windows/Linux server monitoring using Nagios. Objectives: • Perform port and service monitoring using Nagios. • Monitor Linux server resources such as CPU, memory, and disk usage. • Monitor Windows servers using NRPE.	LO6

Textbooks:

1. Rajkumar Buyya, Christian Vecchiola, S ThamaraiSelvi, “Mastering Cloud Computing”, 1st Edition, McGraw-Hill Education, 1st July 2017.
2. Barrie Sosinsky, “Cloud Computing Bible”, Wiley Publishing, 11th January 2011.
3. Ken Cochrane, Jeeva S. Chelladhurai, Neependra Khare, “Docker Cookbook”, Second Edition, Packt publication.
4. Jonathan Baier, “Getting Started with Kubernetes”, Second Edition, Packt Publication.
5. Paul Swartout, DevOps: Continuous Delivery, Integration, and Deployment with DevOps, Pearson Education, First

Edition.
6. Mikael Krief, Learning DevOps – Continuous Delivery, Automation, and Cloud, Packt Publishing, First Edition.

Reference books:

1. Jayaswal, Kailash, Jagannath Kallakurchi, Donald J. Houde, and Deven Shah, Cloud Computing Black Book, 2nd Edition, Dreamtech Press, 1st January 2014.
2. Wittig, Michael, and Andreas Wittig, Amazon Web Services in Action, 2nd Edition, Manning Publications.
3. Kief Morris, Infrastructure as Code: Managing Servers in the Cloud, O'Reilly Media, First Edition.
4. Gene Kim, Jez Humble, Patrick Debois, John Willis, The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, IT Revolution Press, First Edition.

Online References:

1. AWS Documentation https://docs.aws.amazon.com/
2. NPTEL Swayam: https://onlinecourses.nptel.ac.in/noc21_cs14/preview
3. Docker Official Documentation https://docs.docker.com/
4. Kubernetes Official Documentation https://kubernetes.io/docs/
5. Nagios Documentation https://library.nagios.com/products/nagios-core/documentation/

Software Tools

1. Operating System: Ubuntu Linux (Open Source)
2. Containerization Tools: Docker, Docker Compose
3. Container Orchestration: Kubernetes
4. Static Analysis Tool: SonarQube
5. NRPE (Nagios Remote Plugin Executor)

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered).
- The experiments can also be performed using any cloud platform for a similar kind of services.
- Term work will be assessed as a Continuous Internal Assessment Practical (CIAP).

Term work Marks:

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and a minimum passing marks in term work.

Practical Exam: (2 hours/ 25 Marks)

- End-semester Practical / oral exam will be held based on the above syllabus and will be conducted as End Semester Examination Practical (ESEP).

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDML6022	Decision Making and BI Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
MDML6022	Decision Making and BI Lab	--	--	--	25		25

Pre-requisite:

1. CEL304: Skill Lab – Python Programming

Lab Objectives: The course aims to enable students:

1. To build a strong practical foundation in decision-making concepts.
2. To enable learners to use decision analysis tools.
3. To develop hands-on skills in data collection, profiling, cleaning, and transformation.
4. To introduce learners to the fundamentals of BI tools.
5. To guide learners in designing business-oriented KPIs and metrics.
6. To provide experience in developing end-to-end BI solutions.

Lab Outcomes: Upon completion of this course, Learners will be able to:

1. Identify decision types and apply decision-support techniques.
2. Use analytical tools like what-if, goal seek, and sensitivity analysis.
3. Perform data preparation activities.
4. Develop BI reports and create interactive dashboards.
5. Design KPIs, metrics, and visualizations.
6. Construct interactive dashboards and small-scale BI solutions.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	Study of Decision Types & Decision Environments • Identify real-world examples of structured, semi-structured, and unstructured decisions. • Analyse scenarios under certainty, risk, and uncertainty	LO1
2	Implement What-If Analysis using Spreadsheet Tools • Vary input parameters and observe impact on output (e.g., sales forecasting, profit estimation).	LO1

3	Goal Seek Technique for Decision Making • Use spreadsheet tools to determine required input to achieve a desired goal (e.g., break-even point).	LO2
4	Design a Simple Decision Support System (DSS) Model • Build a small DSS model using formulas, functions, and macros for a given business scenario.	LO2
5	Scenario Analysis & Multi-Criteria Decision Making (MCDM) using Spreadsheets • Define a decision problem with 3–5 alternatives and multiple criteria (e.g. cost, time, quality, risk). • Assign weights to criteria. • Use spreadsheet to compute a score for each alternative (weighted sum or simple normalized scoring).	LO2
6	Data Collection and Data Profiling • Collect a dataset (CSV/Excel). • Perform profiling: check data types, missing values, ranges, outliers	LO3
7	Data Cleaning & Transformation • Perform data cleaning (handling missing values, duplicates, formatting). • Apply transformations (normalizing, merging, splitting columns	LO3
8	Design a dashboard that helps a business manager monitor sales performance. Explain which KPIs you would include, how you would structure the layout, and what interactive elements (filters, drill-downs, slicers) you would add to support decision-making.	LO4
9	Creating KPIs and Metrics for BI Reporting • Identify KPIs for a problem statement (e.g., sales, HR, finance). • Design metric calculations in spreadsheet or BI tool.	LO4
10	Designing BI Reports using visualization tools • Import cleaned data. • Create basic reports (tables, cards, simple charts)	LO5
11	Data Integration from Multiple Sources • Combine datasets from different sources (CSV + Excel, or two Excel sheets). • Resolve schema mismatches and perform basic joins using spreadsheet or Power Query	LO5
12	Interactive Dashboard Creation • Create dashboards with slicers/filters. • Add interactive visuals like bar charts, line charts, KPI cards.	LO6
13	Introduction to Visualization – Data Import & Basic Charts • Import data in Tool. • Create bar, line, pie charts, tables. • Work with dimensions & measures.	LO6
14	Cloud-Based BI using Cloud BI • Upload cleaned dataset to cloud storage or connect via web source • Build BI reports/dashboards on cloud BI platform (simulate remote team access)	LO6
15	• BI Case Study • Students prepare a BI solution for any domain:	LO6

	○ Retail ○ HR ○ Healthcare ○ Banking ○ Manufacturing • Includes data preparation, visualization, dashboards, insights, and recommendations.	

Textbooks:

1. Efraim Turban, Ramesh Sharda, Dursun Delen, Decision Support and Business Intelligence Systems, Pearson.
2. James R. Evans, Business Analytics: Methods, Models, and Decisions, Pearson.
3. Nina Godbole, Business Intelligence, Wiley India.
4. Larissa T. Moss & Shaku Atre, Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications, Addison-Wesley.

Reference books:

1. Thomas H. Davenport & Jeanne Harris, Competing on Analytics: The New Science of Winning, Harvard Business Review Press.
2. Carlo Vercellis, Business Intelligence: Data Mining and Optimization for Decision Making, Wiley. (Use BI sections only—avoid ML chapters)
3. Winston W. L., Microsoft Excel Data Analysis and Business Modeling, Microsoft Press.
4. Ponniah P., Data Preparation for Analytics Using SAS, Wiley. (Useful for ETL concepts without OLAP/ML)

Online References:

1. Microsoft Power BI Learning Path https://learn.microsoft.com/power-bi/
2. Tableau Learning Resources https://www.tableau.com/learn
3. Google Data Studio (Looker Studio) Tutorials https://support.google.com/looker-studio

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
MDML6032	Sensor Technology Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
MDML6032	Sensor Technology Lab	--	--	--	25		25

Pre-requisite:

1. FEC2021: Applied Physics-II
2. FEC204: Digital System Design
3. MDMC4031: Microprocessor and Microcontrollers

Program Outcomes Addressed

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solutions.
4. PO5: Engineering Tool Uses
5. PO11: Life-long learning

Lab Objectives: The course aims to enable students:

1. To study fundamental sensor interfacing circuits
2. To study the working principles and characteristics of various sensors and transducers
3. To develop skills to interface microcontroller with various sensors.
4. To implement practical sensor-based systems.
5. To verify sensor circuits to understand behavior, calibration, and performance under different conditions.
6. To apply data processing and basic machine learning techniques.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Construct, test, and validate basic sensor signal conditioning circuits.
2. Measure physical parameters using sensors and plot input–output characteristics.
3. Demonstrate real-time sensing and communication using microcontrollers.
4. Design and implement functional sensor systems, including load-cell force measurement, LDR-based detection circuits, and gas/air-quality monitoring setups.
5. Design and develop automation circuits
6. Simulate sensors and circuits using software tools.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	Basics of the Voltage Divider Circuit and the Wheatstone Bridge: Design and demonstrate how a voltage divider circuit and a Wheatstone bridge can be used to detect and measure small changes in resistance in strain gauges. Analyze which method provides higher sensitivity and justify your choice.	LO1
2	Displacement measurement using an Inductive Transducer (LVDT): Illustrate how an LVDT (Linear Variable Differential Transformer) can be used to measure piston displacement. Examine its working, output characteristics, and suitability for industrial environments.	LO2
3	Characterization of temperature sensors, such as thermocouples, thermistors, and resistance temperature detectors. Compare and evaluate thermocouples, thermistors, and RTDs for this application. Determine which sensor is most appropriate based on accuracy, response time, and stability, and justify your selection with relevant characteristics.	LO2
4	Interfacing Arduino / Raspberry Pi /ESP32 with sensors.: Develop and demonstrate how an Arduino / Raspberry Pi / ESP32 can be interfaced with multiple sensors to collect and transmit environmental data. Illustrate the system architecture and explain data acquisition and communication methods.	LO3
5	Distance measurement using an ultrasonic sensor: Design and implement a system using an ultrasonic sensor to measure distance in real time. Analyze the working principle, calculate distance using echo time, and evaluate factors affecting accuracy in a real-world parking scenario.	LO3
6	Interfacing a Gas Sensor with Arduino/ESP32 for Real-Time Air Quality Monitoring: Design and demonstrate a gas sensor interfaced with an Arduino or ESP32 to monitor air quality in real time. Illustrate the system architecture, including sensor calibration, data acquisition, and wireless transmission. Analyze how the system can be used to trigger alerts when pollutant levels exceed safe thresholds in a real-world urban environment.	LO3
7	Angular velocity measurement using a Gyroscope sensor: Develop and implement a system to measure and analyze vibrations using an accelerometer sensor. Interface an accelerometer with a microcontroller. Record acceleration data along different axes. Convert acceleration signals into vibration parameters (frequency, amplitude). Plot vibration signals and identify patterns. Evaluate system performance under different vibration sources.	LO4
8	Vibration measurement using an Accelerometer: Design a system to measure and analyze vibrations using an accelerometer sensor. Interface an accelerometer with a microcontroller. Record acceleration data along different axes. Convert acceleration signals into vibration parameters (frequency, amplitude). Plot vibration signals and identify patterns. Evaluate system performance under different vibration sources. Display or log the angular velocity readings in real time. Analyze the accuracy of the sensor under different motion conditions.	LO4
9	Develop a sensor-based system to measure force/weight using a load cell: Interface a load cell with an amplifier (e.g., HX711) and microcontroller. Calibrate the system using known weights. Measure unknown forces accurately. Display readings on an LCD or serial monitor. Analyze linearity, sensitivity, and error in measurements.	LO4
10	Design and implement a circuit to detect day and night conditions using a Light Dependent Resistor (LDR):	LO4

	Build a voltage divider circuit using an LDR and a resistor. Set a threshold to distinguish between light and dark conditions. Use a transistor or microcontroller to control an output (LED or relay). Analyze how the resistance of LDR changes with light intensity. Test the circuit under different lighting conditions.	
11	Simulate a voltage divider circuit using Python and verify theoretical and simulated results: Write a Python program to calculate the output voltage for given resistor values. Compare computed results with theoretical calculations. Analyze how the output voltage varies with different resistor combinations.	LO5
12	Introduction to Sensor Simulation Platforms Tools: To understand the fundamentals of sensor interfacing by creating and testing basic sensor circuits using the Tinkercad simulation platform, and to observe how simulated sensor inputs interact with microcontroller-based systems without requiring physical hardware.	LO5
13	Simulated Temperature & Humidity using Measurement Tools: To simulate temperature and humidity measurement using virtual sensors in Tinkercad, interface them with a microcontroller (e.g., ESP 430, Arduino, etc.), and display the sensed temperature and humidity for a weather monitoring station	LO5
14	Simulate Light & Proximity Sensors using Measurement Tools: To design and simulate circuits using light and proximity sensors in Tinkercad, observe sensor response variations with changing environmental conditions, and analyze their role in Automatic Street lighting systems or Touchless switching systems	LO5
15	Sensor Data Processing, Analysis & Noise Filtering Using Python (NumPy, Pandas): To process simulated sensor datasets using Python libraries such as NumPy and Pandas, perform statistical analysis, and apply filtering techniques to reduce noise and improve sensor data accuracy for signal conditions in IOT applications.	LO6
16	Sensor Data Classification using Machine Learning Algorithms: To apply machine learning algorithms for classification of sensor data, train models using labeled datasets, and evaluate their performance in identifying patterns for Activity recognition using wearable sensors or Smart healthcare monitoring systems	LO6

Textbooks:

1. Jacob Fraden, “Handbook of Modern Sensors: Physics, Designs and Applications”, 2016, 5th edition, Springer, New York.
2. John H. Davies, “MSP430 Microcontroller Basics”, 2011, 2nd ed., Newnes Publishing, New York.
3. Jonathan W. Valvano, Embedded Systems: Introduction to Arm Cortex-M Microcontrollers, 2017, 5th edition.

Reference books:

1. Sergey Y. Yurish,” Digital Sensors and Sensor Systems: Practical Design”, 2011, 1st ed., IFSA Publishing, New York.
2. Zach Shelby, Carsten Bormann, "6LoWPAN: The Wireless Embedded Internet", 2009, 1st ed., John Wiley & Sons, New Jersey.
3. Sergey Y. Yurish, Digital Sensors and Sensor Systems: Practical Design, 2011, 1st edition, IFSA Publishing, New York.

Online References:

1. https://www.arduino.cc/en/software/

Software Tools

1. Arduino IDE, Tinkercad, Python / Jupiter

Term Work:

- Term work should consist of at least 10 experiments+ 2 experiments on Simulator (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and a minimum passing mark in term work. Term work will be assessed as a Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiments) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL6011	Machine Vision Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL6011	Machine Vision Lab	--	--	--	25		25

Pre- requisite:

1. CEL304: Skill Lab – Python Programming

Program Outcomes Addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design /Development of Solutions
4. PO4: Conduct Investigations
5. PO6: The Engineer and the world
6. PO11: Life Long Learning.

Lab Objectives: The course aims to enable students:

1. To study and perform digital image acquisition, visualization, and storage using cameras or input devices and implement display operations using OpenCV.
2. To apply geometric and spatial transformations and analyze their impact on image structure and content.
3. To enhance image quality using spatial and frequency domain techniques, including filtering, histogram equalization, smoothing, and sharpening.
4. To implement feature extraction and modeling techniques, including edges, corners, descriptors and evaluate feature matching approaches.
5. To implement and analyze segmentation and detection methods, including thresholding, region/morphological segmentation, classical object detectors, and deep learning-based object detection.
6. To explore advanced and emerging machine vision applications, including 3D imaging, depth sensing, facial/gesture recognition, medical imaging, and trends in Industry 4.0 and smart manufacturing.

Lab Outcomes: Upon completion of this course, Students will be able to:

1 Acquire, visualize, and process digital images using OpenCV and camera interfaces.
2 Apply and evaluate image transformations for pre-processing and alignment in machine vision workflows.
3 Analyze and enhance image quality using denoising, smoothing, sharpening, and contrast-improvement methods.
4 Extract, detect, and match visual features using edge detectors, corner detectors, and local feature descriptors for recognition and tracking tasks.

5	Implement object segmentation and detection pipelines, and quantitatively evaluate results using IoU, precision, recall, F1-score, and mAP.
6	Develop machine-learning and deep-learning-based vision systems, including object classification, detection, and 3D vision applications relevant to modern industrial automation.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	To perform Image acquisition and visualization on case study such as Smart Attendance System • Capture student images using a webcam • Display real-time video feed • Store images in a database for further processing • Analyze image formats and resolution impact	LO1
2	To apply and analyze various geometric and spatial image transformations on case study such as Document Scanner App • Correct skewed document images • Apply scaling, rotation, affine, and perspective transformation • Compare transformation accuracy	LO2
3	To enhance image quality by applying various spatial and frequency domain enhancement techniques on case study such as CCTV Footage system. • Improve low-light images • Apply histogram equalization, filtering, and FFT-based enhancement • Evaluate quality using PSNR/SSIM	LO3
4	To implement and analyze edge detection and corner detection algorithm on case study such as Lane Detection in Roads. • Detect edges using Canny/Sobel • Identify corners for lane marking • Analyze performance in noisy conditions.	LO4
5	To implement Local Feature Descriptor Extraction and Matching techniques on case study such as Panorama Image Stitching • Extract features (SIFT/ORB) • Match features across images • Stitch multiple images into a panorama.	LO4
6	To perform motion tracking in image sequences using feature-based tracking methods on case study such as Traffic Monitoring System • Track moving vehicles across frames • Analyze trajectory and speed.	LO4
7	To implement optical flow techniques on case study such as Crowd Movement. • Compute dense/sparse optical flow • Visualize motion fields • Analyze crowd direction and density.	LO4

8	To segment objects on case study such as Tumor Detection in Medical Images • Segment region of interest • Apply thresholding, region-based, morphology • Evaluate segmentation accuracy (IoU, Dice score)	LO5
9	To implement object detection techniques on case study such as Face Detection Security System • Detect faces using Haar Cascade / HOG • Evaluate detection accuracy	LO5
10	To perform image-based object measurement and dimension estimation on any 3D object • Measure object dimensions using calibration • Use contour detection for shape analysis	LO5
11	To study and implement 3D imaging techniques and depth analysis on case study such as Smart Factory Automation (Industry 4.0) • Generate depth maps / point clouds • Analyze object distance and spatial layout	LO6
12	To explore any modern machine vision applications such as Smart Surveillance System • Implement face/gesture recognition • Explore edge-AI deployment	LO6
13	To perform camera calibration on case study such as Robotics Vision System • Perform camera calibration • Correct lens distortion • Compute intrinsic/extrinsic parameters	LO1, LO2
14	To implement machine learning techniques for image classification on case study such as Waste Classification System • Extract features (HOG/SIFT/ORB) • Apply PCA/LDA • Train classifiers (KNN, SVM, Decision Tree)	LO4, LO5
15	To perform object detection using deep-learning-based architectures on case study such as Autonomous Driving System • Implement YOLO / R-CNN • Detect multiple objects in real-time • Evaluate performance (mAP, FPS)	LO5, LO6

Textbooks:

1. Learning OpenCV 3 Computer Vision with Python Second Edition, by Joe Minichino Joseph Howse Published by Packt Publishing Ltd.
2. R. Szeliski, Computer Vision: Algorithms and Applications, 2nd ed., Springer, 2022.
3. R. Gonzalez and R. Woods, Digital Image Processing, 4th ed., Pearson, 2018.

Reference books:

1. S. Prince, Understanding Deep Learning, MIT Press, 2023.
2. Rosebrock, Deep Learning for Computer Vision, PyImageSearch, 2021.

Online References:

1. Image Processing and Computer Vision Lab http://iitk.ac.in/ee/computer-vision-lab
2. Nptel course on Computer vision and Image Processing https://nptel.ac.in/courses/108103174
3. Open-Source Computer Vision tutorial https://docs.opencv.org/3.4/d9/df8/tutorial_root.html

Term Work:

- Term work should consist of at least 10 experiments+ 2 experiments on Simulator (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and a minimum passing mark in term work. Term work will be assessed as a Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL6012	Robotics and Applications Lab	--	2	--	--	1	--	1

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL6012	Robotics and Applications Lab	--	--	--	25	--	25

Pre- requisites:

1. Basic knowledge of mathematics.
2. FEL104: C Programming

Program Outcomes Addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO5: Engineering tool Usage
5. PO6: The Engineer and the World
6. PO8: Ethics
7. PO11: Life-long Learning.

Lab Objectives: The course aims to enable students:

1. To understand fundamentals of robotics including kinematics, dynamics, and control.
2. To gain hands-on experience with sensors, actuators, and robotic mechanisms.
3. To perform robot simulation using ROS, Gazebo or Webots
4. To implement motion planning, path planning, and mobile robot navigation.
5. To develop robot programming skills using ROS, Python and microcontrollers.
6. To integrate hardware and software systems in a robotic platform.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Apply fundamental and advanced concepts of robot kinematics and dynamics to analyze and solve robotics problems.
2. Design and implement embedded control systems by interfacing sensors and actuators with microcontrollers.
3. Develop and test perception algorithms using vision and sensor data for environment interaction and obstacle avoidance.
4. Formulate and implement path planning and control algorithms for mobile and manipulator robots.
5. Integrate AI techniques and ROS-based software tools for developing and deploying advanced robotic

applications.

6. Develop interdisciplinary problem-solving skills by integrating mechanical design, electronics, programming, and data analysis in robotics projects.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	A robotics startup is setting up its development environment for simulation and prototyping. As a robotics engineer, you are required to install and configure essential tools. **Tasks:** • Install ROS, Gazebo, CoppeliaSim, and required Python libraries. • Verify installation by running sample simulations. • Configure environment variables and dependencies.	LO1
2	A robotic arm needs to accurately move and orient objects in a manufacturing setup. You are required to model its movement mathematically. **Tasks:** • Implement rotation and translation matrices in Python. • Apply transformations to coordinate points. • Visualize transformations in 2D/3D space.	LO1
3	A factory robot must accurately position its end-effector for assembly operations. You need to compute its position based on joint angles. **Tasks:** • Design forward kinematics for a 2-DOF/3-DOF manipulator. • Implement on hardware setup. • Validate end-effector position using measurements and visualization tools.	LO2
4	A robotic arm is required to reach specific target points in space. You must determine joint angles needed to achieve these positions. **Tasks:** • Develop inverse kinematics model. • Implement on hardware setup. • Validate accuracy through real-time visualization.	LO2
5	A warehouse robot must navigate safely while avoiding obstacles in a dynamic environment. **Tasks:** • Use ultrasonic/IR sensors for obstacle detection. • Program real-time avoidance logic. • Test robot performance in different scenarios.	LO3
6	A logistics company wants to automate internal transport using mobile robots. **Tasks:** • Program robot movement using Arduino/ESP32. • Implement motion control (forward, reverse, turn). • Test control accuracy and responsiveness.	LO4
7	An automated sorting system must identify objects based on color for packaging. **Tasks:**	LO4

	• Implement color-based object detection using OpenCV. • Capture real-time video feed. • Detect and track colored objects.	
8	A delivery robot must find the shortest path in a warehouse grid while avoiding obstacles. **Tasks:** • Implement A* algorithm in a 2D grid. • Visualize path planning. • Compare optimal vs non-optimal paths.	LO4
9	An industrial robot must follow predefined paths accurately for material transport. **Tasks:** • Design line-following robot using IR sensors. • Implement P and PD control. • Analyze performance and stability.	LO5
10	A robotics team wants to simulate robot navigation before deploying in real environments. **Tasks:** • Use ROS with TurtleBot simulation. • Implement basic navigation commands. • Analyze robot behavior in simulation.	LO5
11	An automated inspection system needs to identify defective products in real time. **Tasks:** • Implement deep learning model for object recognition. • Use real-time camera input. • Evaluate model accuracy and speed.	LO6
12	A robotic arm must handle varying loads, requiring torque estimation for safe operation. **Tasks:** • Develop dynamic model for 2-DOF arm. • Calculate joint torques. • Analyze effect of load variations.	LO2
13	A small-scale industry wants an automated pick-and-place system for packaging. **Tasks:** • Design servo-based robotic arm. • Program pick-and-place operations. • Test accuracy and repeatability.	LO3
14	A service robot needs to be remotely controlled for flexibility and ease of operation. **Tasks:** • Implement Bluetooth/Wi-Fi control. • Develop mobile interface/app. • Test real-time control and latency.	LO5
15	An autonomous robot must navigate unknown environments such as hospitals or warehouses. **Tasks:** • Implement SLAM using ROS. • Build environment map.	LO6

	• Enable autonomous navigation and localization.	

Textbooks:

1. Fundamentals of Robotics: Applied Case Studies with MATLAB® & Python by Hamid D. Taghirad (CRC Press, 2025).
2. Python Programming Handbook for Robotics Development.

Reference books:

1. Modern Robotics: Mechanics, Planning, and Control by Kevin Lynch and Frank Park (2018).
2. Robot Operating System (ROS) for Absolute Beginners: Robotics Programming Made Easy by Lentin Joseph.

Online References:

1. https://www.mathworks.com/products/robotics.
2. ROS Tutorials and Documentation: https://docs.ros.org.
3. OpenCV Documentation: https://opencv.org.
4. https://github.com/aras-labs/Fundamentals_of_Robotics.
5. https://www.mathworks.com/academia/books/matlab-with-python-debray.html.

Software Tools:

1. Python with robotics libraries (Robotics Toolbox for Python, OpenCV)
2. Robot Operating System (ROS / ROS 2)
3. Gazebo or CoppeliaSim for robot simulation
4. Arduino IDE and ESP32 development environment

Hardware Tools:

1. Microcontroller boards: Arduino Uno, ESP32, Raspberry Pi
2. Servo motors (for robotic arm actuation)
3. Ultrasonic and IR sensors (for obstacle detection and line following)
4. Camera or webcam (for vision-based tasks)
5. Mobile robot chassis/platform or TurtleBot (for ROS experiments)
6. Communication modules: Bluetooth/Wi-Fi modules

Term Work:

- Term work should consist of at least 10 experiments+ 2 experiments on Simulator (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensure satisfactory performance of laboratory work and a minimum passing mark in term work. Term work will be assessed as a Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL6013	Digital Forensics Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL6013	Digital Forensics Lab	--	--	--	25	--	25

Pre- requisite:

1. CEC503: Computer Network
2. CEPEC5013: Ethical Hacking

Program Outcomes Addressed

1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct Investigations of Complex Problems
5. PO5: Engineering Tool Usage
6. PO6: The Engineer and The World
7. PO7: Ethics
8. PO8: Individual and Collaborative Team work
9. PO9: Communication
10. PO11: Life-Long Learning

Lab Objectives: The course aims to enable students:

1. To demonstrate the procedures for identification, preservation, and acquisition of digital evidence.
2. To demonstrate techniques and tools used in digital forensics for operating systems and malware investigation.
3. Ability to perform forensic analysis of hard disks, volatile memory, and malicious software.
4. To demonstrate tools for mobile forensics and browser forensics
5. To explore scenario-based crime forensics investigations.
6. Understanding of social media forensic investigation.

Lab Outcomes: Upon completion of this course, Students will be able to:

1. Explore various forensics tools and use them to acquire, duplicate and analyze data and recover deleted data.
2. Perform preview-based analysis and selective digital evidence acquisition using FTK Imager.
3. Explore various usage of forensics tools to acquire and analyze live/static data.
4. Demonstrate Timeline Report Analysis using forensics tools.
5. Explore and analyze usage of forensics tools on various Operating Systems.

6. Discuss real-time crime forensics investigations scenarios.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	A cybersecurity investigation unit needs to analyze seized digital evidence from a suspect system. The evidence is provided as a forensic disk image. The team must use open-source forensic tools to examine the image, recover critical artifacts, and generate a legally admissible report while maintaining evidence integrity. **Investigation Steps:** a) Identify which tool should be used at different stages. b) Justify the selection based on functionality and performance c) Explain how data flows through these tools in the given setup • Evidence Acquisition • Image Loading • Analysis Phase • Investigation Output • Reporting	LO1
2	A cybersecurity investigation unit needs to acquire and duplicate digital evidence from a suspect storage device. The evidence must be collected in a forensically sound manner to ensure its integrity and admissibility in legal proceedings. The team must use open-source tools available in Kali Linux, to create exact bit-by-bit images of the storage media, verify the integrity of the acquired data using hashing techniques, and maintain proper documentation of the acquisition process. **Investigation Steps:** a) Identify appropriate tools for bit-by-bit image extraction. b) Justify the selection based on functionality and performance c) Explain how data flows through these tools in the given setup o Evidence Acquisition o Image Loading o Analysis Phase o Investigation Output o Reporting	LO1
3	A corporate organization suspects that sensitive data has been exfiltrated using unauthorized USB storage devices. The security team must investigate system artifacts to identify connected USB devices, determine usage patterns, and correlate them with potential data leakage incidents using specialized forensic tools. **Investigation Steps:** a) Identify appropriate tools for USB forensic analysis b) Justify tool selection based on functionality and performance c) Explain forensic workflow • Evidence Collection • Artifact Extraction • Analysis (Unauthorized devices, Repeated usage patterns, Suspicious timestamps etc.) • Correlation (File access logs, User login sessions)	LO1

	• Reporting	
4	A cybercrime investigation unit has seized a suspect system believed to contain confidential data. Due to limited storage and time constraints, the investigators must preview the evidence and selectively acquire only relevant data instead of performing full disk imaging. The process must ensure evidence integrity and forensic soundness. **Investigation Steps:** a) Forensic Tool Identification b) Justification of Selected Approach c) Evidence Handling Workflow • Load Evidence Source • Preview Analysis (Browse file system structure for Suspicious folders, Recently accessed files, Hidden or deleted content) • Selective Acquisition (Export only relevant files/folders, Capture metadata (timestamps, file attributes)) • Integrity Verification • Documentation (Selected evidence, Hash values, Acquisition method)	LO2
5	A security analyst is provided with a memory image captured from a system suspected of compromise. The objective is to analyze volatile memory to uncover traces of malicious activity, including hidden processes, injected code, and attacker presence. **Investigation Steps:** a) Choose suitable tools for memory acquisition and analysis. b) Assess tools based on capability to capture RAM and detect memory-based threats. c) Using the selected tools, acquire RAM and analyze the memory image, to identify system profile, detect suspicious processes, injected code, and network activity, and correlate findings to report indicators of compromise (IoCs)	LO3
6	A cybersecurity investigation unit needs to analyze captured network traffic from a suspected data breach incident. The team must reconstruct communication sessions, extract transferred files, and identify suspicious activities from the data. **Investigation Steps:** a) Identify which analysis techniques (host identification, file extraction, credential analysis, session reconstruction) should be applied at different stages of the investigation. b) Justify the selection of these techniques based on their effectiveness in passive analysis, evidence preservation, and accurate reconstruction of network events. c) Explain how network data is processed during analysis, including packet capture input, protocol parsing, reconstruction of transmitted files/images, and identification of hosts, sessions, and potential evidence.	LO3
7	A security analyst is provided with a suspected malicious file and must perform analysis to determine its behavior, functionality, and potential impact on the system using specialized malware analysis tools. **Investigation Steps:** a) Inspect the file using open-source tools to extract metadata, strings, and hashes. b) Perform reverse engineering with an open source tool to understand code behavior c) Identify indicators of compromise (IoCs) without executing the malware.	LO3

8	A digital forensic investigation unit needs to analyze a disk image obtained from a suspect system to understand user activities over time and reconstruct a sequence of events. **Investigation Steps:** a) Identify which artifacts (file system metadata, logs, browser history, registry entries, and timestamps) should be examined to build an accurate timeline of events. b) Justify the use of timeline analysis based on its ability to correlate multiple data sources, reveal hidden patterns, and provide chronological reconstruction of user and system activities. c) Explain how the timeline is generated, including ingestion of the disk image, extraction of timestamped artifacts, normalization of time data, and visualization of events in chronological order for investigation.	LO4
9	Select a real-world or simulated cyber incident scenario (such as unauthorized data access, suspicious browsing activity, insider threat, or data exfiltration) and perform a timeline and correlation analysis. Using Autopsy, log2timeline (Plaso), and Timesketch, they must extract and correlate browser history, system logs, and file system events to reconstruct the sequence of activities and identify potential security incidents.	LO4
10	A cybersecurity team needs to investigate a suspected security incident on a Windows system by analyzing system and application logs, detecting anomalies, and identifying potential malicious activities. **Investigation Steps:** a) Identify which log sources and techniques (system logs, application logs, security logs, process monitoring, and centralized log analysis) should be used at different stages of the investigation. b) Justify the selection of these approaches based on their effectiveness in real-time monitoring, detailed event tracking, correlation of events, and detection of suspicious behavior. c) Explain how log data is processed during analysis, including log collection, parsing, normalization, correlation of events across sources, and identification of indicators of compromise for incident investigation.	LO5
11	A digital forensic investigation unit needs to recover deleted or hidden files from a storage medium where file system metadata is missing or corrupted. The objective is to extract usable data fragments based on file signatures. **Investigation Steps:** a) Identify which data carving approaches (header-footer based recovery, pattern matching, and file type-specific carving) should be applied at different stages of the recovery process. b) Justify the selection of these approaches based on their ability to recover fragmented or deleted files without relying on file system structures, ensuring efficient and accurate evidence retrieval. c) Explain how data carving is performed, including scanning raw data for file signatures, identifying headers and footers, reconstructing file contents, and extracting recoverable files for further forensic analysis.	LO5

12	A digital forensic investigation requires examination of deleted files from a Windows system to analyze user activity, understand file deletion patterns, and recover potential evidence from the recycle bin. **Investigation Steps:** a) Identify recycle bin artifacts such as deleted file records, metadata, original paths, timestamps, and user identifiers. b) Determine relevant artifacts required to reconstruct file deletion events and user activity. c) Assess the role of recycle bin data in revealing user actions and supporting evidence recovery. d) Analyze recycle bin structures and metadata to correlate file details and reconstruct deletion timelines.	LO5
13	A system is suspected of being used to access unauthorized or malicious websites. Students must analyze browser artifacts stored in database files to reconstruct browsing activity and identify suspicious behavior. The task involves examining history, cookies, and download records to trace user actions and detect potential security incidents. **Investigation Steps:** a) Locate browser database files containing history, cookies, and downloads b) Examine database tables to retrieve URLs, timestamps, and user activity details c) Correlate browsing records with timelines to identify suspicious websites or actions	LO6
14	A digital forensic analyst is provided with a collection of digital files from a simulated investigation. The objective is to extract embedded metadata and perform open-source intelligence (OSINT) exploration to identify the origin, ownership, and relationships associated with the data, and to uncover hidden or sensitive information relevant to the investigation. **Investigation Steps:** a) Extract embedded information such as author details, timestamps, device information, and geolocation from the given files b) Use extracted attributes (names, emails, locations, keywords) to gather related information from publicly available sources c) Correlate metadata and collected intelligence to identify connections, validate findings, and derive investigative insights	LO6
15	A digital forensic analyst is assigned to profile a publicly available social media user as part of an investigation. The objective is to gather and analyze publicly accessible information using search engines and open-source intelligence techniques to identify the user's digital footprint, associations, and behavioral patterns. **Investigation Steps:** a) Gather publicly available information from social media profiles, search results, and related online sources. b) Identify key details such as usernames, connections, locations, interests, and activity patterns c) Correlate collected data to build a comprehensive user profile and identify relationships or potential risks	LO6

Textbooks:

1. J. T. Luttgens, M. Pepe, and K. Mandia, Incident Response & Computer Forensics, 3rd ed. New York, NY, USA: McGraw-Hill Professional, 2014.
2. N. A. Hassan, Digital Forensics Basics: A Practical Guide Using Windows OS. Berkeley, CA, USA: Apress, 2019.
3. X. Lin, Introductory Computer Forensics: A Hands-on Practical Approach. Cham, Switzerland: Springer Nature, 2018.
4. J. Golbeck, Introduction to Social Media Investigation: A Hands-on Approach. Syngress, 2015.

Reference books:

1. M. Ligh, A. Case, J. Levy, and A. Walters, The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory. Hoboken, NJ, USA: Wiley, 2014.
2. T. S. Ho and S. Li, Eds., Digital Forensics of Multimedia Data and Devices. Hoboken, NJ, USA: Wiley, 2015.
3. J. R. Vacca, Ed., Computer and Information Security Handbook, 3rd ed. Amsterdam, Netherlands: Elsevier, 2017.
4. S. Bommisetty, R. Tamma, and H. Mahalik, Practical Mobile Forensics, 4th ed. Birmingham, U.K.: Packt Publishing, 2023.
5. N. Jain and D. Kalbande, Digital Forensics. Hoboken, NJ, USA: Wiley, 2023.

Online References:

1. TryHackMe – "Virtual Labs for Learning Digital Forensics"
2. Packt / Coursera — "Digital Forensics for Pentesters: Hands-On Learning"

Software Tools

1. FTK Imager(Windows/Linux)
2. Guymager(Linux)
3. DumpIt(Capture RAM)
4. Autopsy+Sleuth Kit
5. FTK Tool kit
6. Scalpel
7. Wireshark
8. MFCMAPI
9. Timesketch

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered.)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEPEL6014	Natural Language Processing Lab	--	02	--	--	01	-	01

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEPEL6014	Natural Language Processing Lab	--	--	--	25	--	25

Pre- requisite:
1. CEL304: Skill Lab – Python Programming

Program Outcomes Addressed
1. PO1: Engineering Knowledge
2. PO2: Problem Analysis
3. PO3: Design/Development of Solutions
4. PO4: Conduct Investigations of Complex Problems
5. PO5: Engineering Tool Usage
6. PO11: Life-Long Learning

Lab Objectives: The course aims to enable students:
1. To understand text preprocessing techniques for real-world NLP tasks.
2. To familiarize with linguistic analysis through morphology, N-grams, and chunking.
3. To build statistical analysis for N-grams and parsing.
4. To understand semantic analysis and document similarity techniques
5. To learn NLP systems using standard classification techniques.
6. To understand large language models for understanding architectural variations.

Lab Outcomes: Upon completion of this course, Students will be able to:
1. Apply various text processing techniques for real-world NLP tasks..
2. Analyze linguistic structures using morphological processing, N- grams, and chunking techniques.
3. Develop statistical and neural language models such as POS tagging, HMMs, and Parsers.
4. Use TF-IDF and cosine similarity to extract semantics and document distances
5. Analyze the performance of modern NLP models including NER, and BERT classification.
6. Compare outputs and capabilities of modern large language models to understand architectural differences.

Suggested List of Experiments: Students are required to complete at least 12 experiments.

Sr. No.	Title of Experiments	LO Mapped
1	Basic Text Preprocessing Techniques **Task:** You are given raw, unstructured text data and need to clean and prepare it for analysis. Using Python in Jupyter Notebook with NLTK, spaCy, and regex, you will apply basic preprocessing techniques like tokenization, filtering, script validation, stopword removal, stemming, and lemmatization. The aim is to convert noisy text into meaningful and structured tokens for further processing. **Tools:** Python, Jupyter Notebook, NLTK, spaCy, regex..	LO1
2	Perform morphological analysis on input text to extract root words and morphemes, and generate new word forms using inflectional and derivational rules. Input is eaw text and output is root forms, morphemes, and generated word variants. Use stemming (e.g., Porter Stemmer in NLTK) for fast root extraction, or lemmatization (spaCy) for accurate dictionary forms. Tools: Python, Jupyter Notebook, NLTK, spaCy, morphological analyzers.	LO1
3	Extraction of Email Addresses and Indian Mobile Numbers Using Regular Expressions **Task:** Write a Python program using Regular Expressions to identify and extract all valid email addresses and Indian mobile numbers from an unstructured text document containing mixed information such as names, dates, and random numbers. **Tools:** Python, Jupyter Notebook or any Python IDE, re (Regular Expressions) module.	LO2
4	Implementation of N-Gram Language Model **Tasks:** You are developing a simple language model to understand how words appear in sequence within a text. Using a given dataset, you will generate unigrams, bigrams, and trigrams, and calculate their probabilities. This helps in predicting the next word and understanding basic language patterns. **Tools:** Python, Jupyter Notebook, NLTK, NumPy.	LO2
5	Evaluation of N-Gram Models Using Laplace and Good-Turing Smoothing Tasks: Apply Laplace and Good-Turing smoothing techniques to N-gram language models and evaluate their impact on text prediction accuracy using appropriate metrics. Tools: Python, Jupyter Notebook, NLTK, NumPy.	LO2
6	Perform chunking (shallow parsing) on text to extract phrases like noun phrases (NP) and verb phrases (VP) **Tasks:** Perform chunking on the given text input to identify and extract syntactic phrases such as noun phrases and verb phrases. Example: “The quick brown fox jumps” → [NP: The quick brown fox] [VP: jumps] **Tools:** Python, Jupyter Notebook, NLTK, spaCy.	LO2
7	Study of Part-of-Speech Taggers and POS Tagging **Tasks:** Study different types of POS taggers (rule-based, statistical) and perform Part-of-Speech tagging on given text. Implement a simple rule-based POS tagger by hardcoding at least 5 grammar rules (e.g., words ending with “-ing” → Verb, “-ly” → Adverb, “-ness” →	

	Noun, capitalized words → Proper Noun, determiners like "the/a" → Determiner) and apply them along with NLTK/spaCy tagging. **Tools:** Python, Jupyter Notebook, NLTK, spaCy.	
8	Demonstration of Viterbi Algorithm for HMM-Based POS Tagging **Tasks:** Demonstrate the Viterbi algorithm for HMM-based POS tagging by first computing transition and emission probability tables from a given dataset, and then using them to find the most probable tag sequence for a sentence. **Tools**: Python, Jupyter Notebook, NumPy, NLTK.	LO2
9	Word Sense Disambiguation Using Lesk Algorithm **Task:** Select a sentence containing an ambiguous target word (e.g., bank). Retrieve all possible senses of the word using WordNet and extract their definitions. Compare the context words of the sentence with each sense's gloss to compute overlap. Choose the sense with the maximum overlap as the correct meanin **Tools:** Python, NLTK.	LO3
10	Word Embeddings Using Word2Vec and GloVe **Tasks:** Generate word embeddings using CBOW and Skip-gram (Word2Vec) and GloVe, and visualize them using dimensionality reduction (e.g., PCA/t-SNE). Analyze semantic relationships by checking similarity between words (e.g., king – man + woman ≈ queen) and observe clustering of related words. **Tools:** Python, Gensim, NumPy	LO3
11	TF-IDF and Document Similarity Analysis **Tasks:** Convert text documents into numerical vectors using TF-IDF by calculating term frequency (TF), inverse document frequency (IDF), and TF-IDF scores. Compute similarity between documents using measures like cosine similarity to identify how closely related the documents are. **Tools:** Python, scikit-learn, NumPy	LO3
12	Study of Transformer Architecture and Self-Attention Mechanism **Task**: Study Large Language Models (LLMs) and explore the core components of the Transformer model, including self-attention, positional encoding, and encoder–decoder architecture using sample text data. **Tools:** Python, Jupyter Notebook, NumPy, PyTorch/TensorFlow, HuggingFace Transformers.	LO4
13	Text Classification Using BERT Fine-Tuning **Tasks:** Fine-tune a pretrained BERT model on a labeled text dataset to perform text classification and evaluate its performance using appropriate metrics. **Tools**: Python, Jupyter Notebook, HuggingFace Transformers, PyTorch/TensorFlow, scikit-learn, datasets library.	LO5
14	Named Entity Recognition (NER) on Text Data	LO5

	Tasks: Implement NER to identify and classify entities such as person names, locations, organizations, dates, and numerical values from input text. Apply pretrained models (NLTK/spaCy) on a sample dataset and analyze the extracted entities. **Tools:** Python, Jupyter Notebook, NLTK or spaCy, pretrained NER models, sample text dataset	
15	Exploration of Modern NLP Models (GPT-4, Llama-3, Claude-3, Mistral, Gemini) **Task:** Compare outputs of modern LLMs on the same text prompt and analyze differences in responses without coding complex models. **Tools:** Web access to LLM APIs or interfaces, Python/Jupyter Notebook (optional for logging results).	LO6

Textbooks:

1. Joseph Babcock, Raghav Bali, "Generative AI with Python and TensorFlow", Packt Publishing, 2021.
2. Daniel Jurafsky, James H. and Martin, Speech and Language Processing, Third Edition, Prentice Hall, 2026.
3. Christopher D.Manning and HinrichSchutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.
4. Pushpak Bhattacharyya and Aditya Joshi, Natural Language Processing, Wiley India Pvt. Ltd., Edition: 2023.
5. Dr. Tanmoy Chakraborty, “Introduction to Large Language Models: Generative AI for Text” , Wiley (Wiley India). December 2024.

Reference books:

1. Daniel M Bikel and ImedZitouni — Multilingual natural language processing applications: from theory to practice, IBM Press, 2013.
2. Palash Goyal, Sumit Pandey, Karan Jain. Deep Learning for Natural Language Processing-2018

Online References:

1. Virtual lab: https://nlp-iiith.vlabs.ac.in/List%20of%20experiments.html
2. Stanford CS224N https://web.stanford.edu/class/cs224n/
3. NLTK https://www.nltk.org/
4. spaCy https://spacy.io/
5. Kaggle https://www.kaggle.com/

Term Work:

- Term work shall consist of at least 12 experiments (Ensure that all the LOs are covered).
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

Term work Marks (CIAP):

- 25 Marks (Total Marks) =20 Marks (Experiment) + 05 Marks (Attendance)
- The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.

SIES Graduate School of Technology
Department of Computer Engineering
Bachelor of Engineering

Course Code	Course Name	Credits Assigned			
		Theory	Practical	Tutorial	Total
CEP601	Major Project I	--	2	--	2

Course Code	Course Name	Examination Scheme					
		Theory Marks			CIAP	ESEP	Total
		Course Assessment		ESE			
		ISE	MSE				
CEP601	Major Project I	--	--	--	25	25	50

Program Outcomes addressed:

1. PO1: Engineering knowledge.
2. PO2: Problem Analysis.
3. PO3: Design/Development of Solutions.
4. PO4: Conduct Investigations of Complex Problems.
5. PO5: Engineering Tool Usage.
6. PO6: The Engineer and The World.
7. PO7: Ethics.
8. PO8: Individual and Collaborative Team work.
9. PO9: Communication.
10. PO10: Project Management and Finance.
11. PO11: Life-Long Learning.

Project Objectives: The course aims to enable students:

1. Investigate and evaluate prominent literature to come with application-oriented project topics in connection with the curriculum.
2. Study and develop an outline for thinking and practice that illuminates and brings insight to the design and implementation aspects with respect to the project topic.
3. Design and create practical resources and solution aspect for the design and implementation.
4. Present an organised exploratory framework, while understanding the documentary deliverables within established academic practices and/ or ideas.
5. Offer inquiry-based argumentation / presentation along with project implementation
6. To promote self-learning and research-oriented thinking.

Project Outcomes: Upon completion of this course, Students will be able to:

1. Perform extensive Review of Literature from diverse knowledge banks or through interactions with Industry experts.
2. Developing or Creating ideas capable of addressing industrial or social solutions to identified problem domains.
3. Acquire knowledge of tools & technologies and application of their expertise in creating project implementation and deliverables.
4. Implement solutions using appropriate technologies, engineering practices, and version control.

5. Test and validate the system through structured test cases, datasets, and simulations.
6. Compose technical documentation, present project outcomes, and demonstrate teamwork, ethics, and self-learning.

1. Guidelines for Project topic selection Process to be defined and followed:

1.1 General Guidelines

a. Project orientation can be given at the end of fifth semester.
b. Students should be informed about the domain and domain experts whose guidance can be taken before selecting projects.
c. Students can select the problem statement from the Industry and provide technology solution as capstone project. Capstone projects in engineering study are considered important as it allow students to integrate and apply the knowledge and skills acquired throughout their academic program and effectively demonstrating their learning of programme by tackling a real-world problem, ultimately keeping them well prepared for the job market
d. Student‘s should be recommended to refer papers from reputed conferences/ journals like IEEE, Elsevier, ACM etc. which are not more than 3 years old for review of literature.
e. Students can certainly take ideas from anywhere, but be sure that they should evolve them in the unique way to suit their project requirements. Students can be informed to refer Digital India portal, SIH portal or any other hackathon portal for problem selection.

1.2 Topics can be finalized with respect to following criterion:

a. Topic Selection: The topics selected should be novel in nature (Product based, Application based or Research based) or should work towards removing the lacuna in currently existing systems.
b. Technology Used: Use of latest technology or modern tools can be encouraged.
c. Students should not repeat work done previously (work done in the last three years).
d. Project work must be carried out by the group of at least 2 students and maximum 4.
e. The project work can be undertaken in a research institute or organization/Industry/any business establishment. (out-house projects)
f. The Capstone Project- problems must be related to the programme or may be interdisciplinary, based on the industry expected outcomes. The individual students have different aptitudes and strengths. Project work, therefore, should match the strengths of students. For this purpose, students should be asked to identify the type of project work they would like to execute.
g. Project shall address National Thrust area such as Environment, Digitization, Automation, sustainability and predefined Sustainable development goals(SDGs).
h. The project proposal presentations can be scheduled according to the domains and should be judged by faculty who are expert in the domain.
i. Head of department and senior staff along with project coordinators will take decision regarding final selection of projects.
j. Guide allocation should be done and students have to submit weekly progress report to the internal guide.
k. Internal guide has to keep track of the progress of the project and also has to maintain attendance report. This progress report can be used for awarding term work marks.
l. In case of industry/ out-house projects, visit by internal guide will be preferred and external members can be called during the presentation at various levels.

1.3 Deployment Requirement

a. Each project must include a deployment phase, ensuring the solution is usable in a real-world or simulated environment.

Examples of deployment include:

- Web application hosted on cloud/server
- Mobile application (APK/Test version/Play Store)
- Industrial automation system implemented or simulated
- AI/ML model deployed via API or dashboard
- IoT system with real-time monitoring and control

2. Guidelines for Assessment of Major Project I:

1.1 Term Work Evaluation (CIAP)

a. A Review/Progress Monitoring Committee shall be constituted by the Head of Department.
b. **Major Project I** shall be evaluated continuously, with a minimum of two reviews per semester.
c. Continuous assessment should also emphasize individual performance, including each student's contribution, understanding of the project, and responses during review interactions.
d. Project report to be submitted as per the guidelines issued by the department.
e. Term work will be assessed as Continuous Internal Assessment Practical (CIAP).

2.2 Distribution of marks

The distribution of Term Work marks for the semester shall be as follows:

a. Marks by Guide/Supervisor based on Logbook: 10
b. Marks by Review Committee: 10
c. Quality of Project Report: 05

1.3 Evaluation Rubrics CIAP (Term work);

a. Quality of literature survey and need identification
b. Clarity of problem definition
c. Innovation and creativity
d. Feasibility and justification of approach
e. Cost effectiveness and societal/environmental impact (SDGs)
f. Quality of system design and use of engineering norms
g. Implementation quality and functioning of prototype
h. Effective use of technical skill sets and tools
i. Teamwork, leadership, and individual contribution
j. Clarity in documentation, presentation, and communication
k. Deployment readiness and implementation feasibility
l. Usability, scalability, and real-world applicability of the deployed solution

1.4 ESEP (Practical/Oral) Examination:

a. **Major Project I** shall be assessed through a presentation and demonstration of working model by the student project group to a panel of Internal and External Examiners preferably from industry or research organizations having experience of more than five years approved by the head of Institution.
b. Students shall be motivated to publish a paper based on the work in Conferences/students competitions.
c. End-semester Practical / oral exam will be held based on the above syllabus and will be conducted as End

Semester Examination Practical (ESEP).

1.5 Evaluation Rubrics ESEP (Practical / Oral):

a. Problem identification and clarity
b. Innovation in solution
c. Feasibility and societal impact
d. Working model functionality
e. Effective use of engineering tools
f. Adherence to standards and norms
g. Teamwork and individual contribution
h. Communication skills
i. Deployment quality (hosting, execution, usability, and performance)