Bharati Vidyapeeth

(Deemed to be University) Pune, India

College of Engineering, Pune

B.Tech. Computer Science & Engineering

(2023 Course)

Program Curriculum As Per NEP Guidelines

VISION OF UNIVERSITY:

Social Transformation through Dynamic Education

MISSION OF UNIVERSITY:

- To make available quality education in different areas of knowledge to the students as per their choice and inclination.
- To offer education to the students in a conducive ambience created by enriched infrastructure and academic facilities in its campuses.
- To bring education within the reach of rural, tribal and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities.
- To make available quality education to the students of rural, tribal and other deprived sections of the population.

VISION OF THE INSTITUTE:

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF THE INSTITUTE:

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.
- Toprovide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions

VISION OF THE DEPARTMENT

To be focused on innovative and quality education in computer science and engineering that prepares professionals for development of society.

MISSION OF THE DEPARTMENT

- To provide academic environment for the development of skilled professionals
- To cultivate research culture that contributes to the sustainable development of the society.
- To enhance academic and industry collaborations for global exposure.

PROGRAM EDUCATIONAL OBJECTIVES

The students of B.TECH. (Computer Science and Engineering), after graduating will able to,

1. Demonstrate technical and professional competencies by applying Engineering Fundamentals, knowledge of computing and technologies.
2. Exhibit effective personality, good communication and team building skills
3. Adopt to the latest trends in the field of computer science and engineering.

PROGRAM SPECIFIC OUTCOMES

1. To design, develop and implement computer programs on hardware towards solving problems.
2. To employ expertise and ethical practice through continuing intellectual growth and adapting to the working environment.

PROGRAM OUTCOMES

a. Apply the knowledge of mathematics, science, engineering, and computing to provide a solution of complex engineering problems.

b. Identify, analyse complex engineering problems to derive conclusions using computer science and engineering knowledge.

c. Outline resolutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration, societal, and environmental considerations.

d. Use existing research knowledge and research techniques including design of experiments, data analysis, and synthesis to provide valid inferences.

e. Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.

f. Apply inferences obtained by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the subsequent responsibilities relevant to the professional engineering practice.

g. Recognize the impact of the professional engineering solutions in societal and environmental contexts to demonstrate the knowledge for sustainable development.

h. Apply ethical principles and execute professional ethics and responsibilities and norms of the engineering practice.

i. Work effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary surroundings.

j. Talk effectively on complex engineering activities with the engineering community such as being able to comprehend and write effective reports and design documentation, make effective presentations.

k. Prove knowledge and understanding of the engineering and management principles and apply these to one’s work, as a member and leader in a team.

l. Recognise the need for and have the preparation and ability to engage in independent and life-long learning in context of technological change.

CORELATION BETWEEN GRADUATE ATTRIBUTES AND PROGRAMME OUTCOMES

Graduate Attributes/ Programme Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and teamwork									✓			
Communication										✓		
Project management and finance											✓	
Life-long learning												✓

A. DEFINITION OF CREDITS:

1 Hour Lecture (L) per week	1 credit
1 Hour Tutorial (T) per week	1 credit
2 Hour Practical (P) per week	1 credit
4 Hours Practical (P) per week	2 credit

B. Course Code and Definition

C. Course Code	Definitions
L	Lecture
T	Tutorial
P	Practical
TW	Term Work
O	Oral
SEE	Semester End Examination
MJ	Major (Core) Course
MI	Minor Course
PE	Professional Elective Course
GE	General Elective Course
OE	Open Elective Course
SE	Skill Enhancement Course
AE	Ability Enhancement Course
VE	Vocational Enhancement Course
VS	Vocational Skill Course
PS	Professional Skill Course
VA	Value Added Course
CC	Co-curricular Courses
EC	Extra-Curricular Course
ID	Inter-disciplinary Course
MD	Multidisciplinary Course
RP	Research I Project Course
PC	Practical Course
BS	Basic Science
ES	Engineering Science
AC	Audit Course
EC	Extracurricular Activities
BM	Basic Mathematics
BP	Basic Physics
BC	Basic Chemistry
UH	Universal Human Values

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Semester – I(NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	OR	PR	Total	Th	Pr/ Or	Tut	Total
1.	BM	BM1113101	Engineering Mathematics- I	3	-	1	60	40	-	-	-	100	3	0	1	4
2.	BC	BC1113102	Engineering Chemistry	3	2	-	60	40	50	-	-	150	3	1	0	4
3.	ES	ES1108103	Digital Electronics	4	2	-	60	40	25		-	125	4	1	0	5
4.	MJ	MJ1113104	Probability and Statistics	4	2	-	60	40	25	-	-	125	4	1	0	5
5.	MJ	MJ1104105	Programming and Problem Solving	4	2	-	60	40	25	-	25	150	4	1	0	5
6.	AE	AE1113106	Communication Skills	-	2	-	-	-	50	-	-	50	0	1	0	1
7.	SE	SE1111107	Skill Base Course -I (Computer-Aided Drawing & Design)	-	2	-	-	-	25	25	-	50	0	1	0	1
			Total	**18**	**12**	**1**	**300**	**200**	**200**	**25**	**25**	**750**	**18**	**6**	**1**	**25**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Semester – II (NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	OR	PR	Total	Th	Pr/ Or	Tut	Total
1.	BM	BM1113201	Engineering Mathematics- II	3	-	1	60	40	-	-	-	100	3	0	1	4
2.	BP	BP1113202	Engineering Physics	3	2	-	60	40	50	-	-	150	3	1	0	4
3.	ET	ET1107203	Electrical Technology	4	2	-	60	40	25	-	-	125	4	1	0	5
4.	MJ	MJ1104204	Discrete Mathematical Structures	4	-	1	60	40	25	-	-	125	4	0	1	5
5.	MJ	MJ1104205	Linear Data Structures	4	2	-	60	40	25	-	25	150	4	1	0	5
6.	UH	UH1113206	Universal Human Values	-	2	-	-	-	50	-	-	50	0	1	0	1
7.	SE	SE1111207	Skill Base Course-II (Computer Workshop Technology)	-	2	-	-	-	25	25	-	50	0	1	0	1
			Total	**18**	**10**	**2**	**300**	**200**	**175**	**25**	**25**	**750**	**18**	**5**	**2**	**25**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Semester – III (NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks							Credits		
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	L	Pr/ Or	T	Total
1.	MJ	MJ1104301	Database Systems	3	2	-	60	40	25	25	-	150	3	1	0	4
2.	MJ	MJ1104302	Non-Linear Data Structures	3	2	-	60	40	25	25	-	150	3	1	0	4
3.	MJ	MJ1104303	Software Engineering	3	-	1	60	40	25	-	-	125	3	-	1	4
4	MJ	MJ1104304	Machine Organization and Microprocessors	3	-	-	60	40	-	-	-	100	3	-	0	3
5.	MJ	MJ1104305	Object Oriented Methodology	3	2	-	60	40	25	-	-	125	3	1	-	4
6.	SE	SE1104306	Skill Base Course-III (Computer Skill Lab- I) Web Programming	-	2	-	-	-	25	25	-	50	0	1	0	1
			Total	**15**	**8**	**1**	**300**	**200**	**125**	**75**	**-**	**700**	**15**	**4**	**1**	**20**
7.	Audit Course-I	AC1113307	Indian Knowledge System	2	-	-	-	100	-	-	-	-	-	-	-	2
8.	*Value Added Course	VA1104308	VAC- I	2	-	-	-	100	-	-	-	-	-	-	-	2

***Indicate this is mandatory but the credits will not be considered in SGPA/CGPA.(As and when the students complete the course and submit the certificate, it should be reflected in the mark sheet. The student should clear the subject up to 7th Sem of his/her coursework.)**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Semester – IV (NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	L	Pr/ Or	T	Total
1.	MJ	MJ1104401	Theory of Computing	3	-	1	60	40	25	-	-	125	3	0	1	4
2.	MJ	MJ1104402	System Programming and Operating Systems	3	2		60	40	25		25	150	3	1	-	4
3.	MJ	MJ1104403	Computer Organization and Architecture	3	-	-	60	40	-	-	-	100	3	0	0	3
4.	MJ	MJ1104404	Computer Graphics and Multimedia	3	2	-	60	40	25	-		125	3	1	0	4
5.	MJ	MJ1104405	Computer Networks	3	2	-	60	40	25	25		150	3	1	-	4
6.	SE	SE1104406	Skill Base Course-IV (Python)	-	2	-	-	-	25	25		50	0	1	0	1
			Total	**15**	**8**	**1**	**300**	**200**	**125**	**75**	**25**	**700**	**15**	**4**	**1**	**20**
7.	*MOOC/ Swayam NPTEL	AE1104407	MOOC-I	-	-	-	-	-	-	-	-	-	-	-	-	2
8	*Social Activity	EC1104408	Social Activity	-	-	-	-	-	-	-	-	-	-	-	-	2

*** Indicate this is mandatory but the credits will not be considered in SGPA/CGPA.(As and when the students complete the course and submit the certificate, it should be reflected in the mark sheet. The student should clear the subject up to 7th Sem of his/her coursework.)**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE B. Tech. (CSE): Semester – V (NEP 2020 COURSE)

Sr. No.	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	MJ	MJ1104501	Design and Analysis of Algorithm	3	2	-	60	40	25	25	-	150	3	1	-	**4**
2.	MJ	MJ1104502	Artificial Intelligence	3	2	-	60	40	25	-	25	150	3	1	-	**4**
3.	MJ	MJ1104503	Design Thinking	3	-	-	60	40	-	-	-	100	3	-	-	**3**
4.	MJ	MJ1104504	Internet of Things	3	-	-	60	40	-	-	-	100	3	-	-	**3**
5.	PE	PE1104505	Course – 5 (PEC- I)	3	2	-	60	40	25	-	25	150	3	1	-	**4**
6.	SE	SE1104506	Skill based Course –V (Advance Java Programming)	-	4	-	-	-	25	25	-	50	-	2	-	**2**
			Total	**15**	**10**	-	**300**	**200**	**100**	**75**	**25**	**700**	**15**	5	-	**20**
8.	VA	VA1104507	**Value Added Course- II	-	2	-	-	40	-	-	-	-	-	2	-	2
9.	AE	AE1104508	MOOC - II	-	-	-	-	-	-	-	-	-	-	-	-	2

Program Elective Course (PEC)

Semester-V
PEC-I
Image Processing
Software Testing and Quality Assurance
Compiler Design

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY) COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Semester – VI (NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/ Or	Tut	Total
1.	MJ	MJ1104601	Machine Learning	3	2	-	60	40	25	25	-	150	3	1	-	**4**
2.	MJ	MJ1104602	Enterprise Software Development	3	2	-	60	40	25	25	-	150	3	1	-	**4**
3.	MJ	MJ1104603	Cloud Computing	3	-	-	60	40	-	-	-	100	3	-	-	**3**
4.	MJ	MJ1104604	Natural Language Processing	3	-	-	60	40	-	-	-	100	3	-	-	**3**
5.	PE	PE1104605	Course – 5 (PEC-II)	3	2	-	60	40	25	-	-	125	3	1	-	**4**
6.	PS	PS1104606	Professional Skills	-	2	-	-	-	25	-	-	25	-	1	-	**1**
7.	SE	SE1104607	Skill based Course–VI (GenAI)	-	2	-	-	-	25	-	25	50	-	1	-	**1**
			Total	**15**	**10**		**300**	**300**	**125**	**50**	**25**	**800**	**-**	**-**	**-**	**24**
7.	AC	AC1104608	Environmental Studies	4	-	-	-	100	-	-	-	100	4	-	-	**4**

Program Elective Course (PEC)

Semester -VI
PEC-II
Data Mining and Analytics
Computer Security and Digital Forensics
Business Intelligence

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY) COLLEGE OF ENGINEERING, PUNE

TRACK-I

B.Tech.(Computer Science and Engineering): Semester–VII(2023 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	MJ	MJ1104701	Deep Learning	03	02	-	60	40	25	25	-	150	03	01	-	04
2.	PE	PE1104702	Program Elective Course-III	03	02	-	60	40	25	-	25	150	03	01	-	04
3.	PE	PE1104701	Program Elective Course-IV	03	02	-	60	40	25	-	25	150	03	01	-	04
4.	RP	RP1104704	Major Project	-	16	-	-	-	100	50		150	-	08	-	08
			Total	**09**	**22**	**-**	**180**	**120**	**175**	**75**	**50**	**600**	**09**	**11**	**-**	**20**

Professional Elective Course (PEC) List

Programme Elective Course-III	Programme Elective Course-IV
Robotic Process Automation	Blockchain Technology
AR-VR	IT Project Management (DevOPS)
Information Storage and Retrieval	Agentic AI

B.Tech.(Computer Science and Engineering):Semester–VIII (2023COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
1.	MJ	MJ1104801	Computer Vision	3	2	-	60	40	25	-	25	150	03	01	-	04
2.	CC	CC1104802	Seminar	-	4	-	-	-	50		50	100	-	02	-	02
3.	SE	SE1104803	Internship(industry/Inhouse)	-	-	-	-	-	150	-	100	250	-	14	-	14
			Total	**3**	**6**	**-**	**60**	**40**	**225**	**-**	**175**	**500**	**03**	**17**	**-**	**20**

TRACK-II

B.Tech.(Computer Science and Engineering):Semester–VII (2023COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
4.	MJ	MJ1104801	Computer Vision	3	2	-	60	40	25	-	25	150	03	01	-	04
5.	CC	CC1104802	Seminar	-	4	-	-	-	50		50	100	-	02	-	02
6.	SE	SE1104803	Internship(industry/Inhouse)	-	-	-	-	-	150	-	100	250	-	14	-	14
			Total	**3**	**6**	**-**	**60**	**40**	**225**	**-**	**175**	**500**	**03**	**17**	**-**	**20**

B.Tech.(Computer Science and Engineering):Semester–VIII (2023 CBCS COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	Th	Pr/Or	Tut	Total
5.	MJ	MJ1104701	Deep Learning	3	2	-	60	40	25	25	-	150	03	01	-	04
6.	PE	PE1104702	Program Elective Course-III	3	2	-	60	40	25	-	25	150	03	01	-	04
7.	PE	PE1104701	Program Elective Course-III	3	2	-	60	40	25	-	25	150	03	01	-	04
8.	RP	RP1104704	Major Project	-	16	-	-	-	100	50		150	-	08	-	08
			Total	**9**	**22**	**-**	**180**	**120**	**175**	**75**	**50**	**600**	**09**	**11**	**-**	**20**

Professional Elective Course (PEC) List

Programme Elective Course-III	Programme Elective Course-III
Robotic Process Automation	Blockchain Technology
AR-VR	IT Project Management (DevOPS)
Information Storage and Retrieval	Agentic AI

Instructions

1. Students shall be permitted to opt for either Track-1 or Track-2.
   A. If the student opts Track-1 then he/she has to perform Major Project in Semester-VII and undergo Internship in Semester-VIII.
   B. If the student opts Track-2 then he/she has to undergo Internship in Semester-VII and perform Major Project in Semester-VIII.

2. Seminar:

Objectives of the Seminar

The Seminar aims to:

- Develop self-learning ability and research orientation
- Enhance technical understanding of emerging technologies
- Improve presentation and communication skills

Scope of Topics

Students must select a topic from **Emerging/Recent Technologies, Case Studies, Research-Oriented Topics, Interdisciplinary Topics**

What Students Are Expected to Do

Topic Selection

Literature Survey

Content Preparation

Report Writing

Presentation

3. **Major Project:**

Objectives of the Major Project Students are expected to: • Apply knowledge of Computer/IT engineering to solve real-world problems • Develop design, implementation, and testing skills • Work effectively in a team environment • Demonstrate innovation, problem-solving, and research aptitude • Learn documentation, presentation, and communication skills • Get exposure to research methodology and publication process Nature of Projects Projects should be: Application-oriented / Industry-based / Research-based / Interdisciplinary Should NOT be repetitive of previous batches Must include significant development and/or research contribution Research Publication To encourage the students for writing research article/paper and presenting/publishing the same in journal/conference/symposium, the guides will may give advantage of credits to the students, even if it is not mandatory. 4. Internship Objectives of Internship • To provide real-world exposure to professional environments • To enhance technical, problem-solving, and teamwork skills • To bridge the gap between academics and industry expectations • To develop professional ethics, communication, and responsibility Guidelines 1) The students who receive internship in the industry (through institute or own efforts), will do it for one complete semester in the industry. 2) Those students who don not receive the internship in the industry, will have to perform it in-house. The Department will prepare modules for in-house internship based on recent domains/technology. The Department faculty plus external experts from industry fir the selected domain will deliver the sessions. In this case, the entire internship of one complete semester will be held in the institute itself. 3) The projects and the required computational facilities for these projects are available at FAIR Lab. The students may use facilities in IDEA and Intel Unnati Lab for this purpose.

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Minor Degree Course: Blockchain (NEP 2020 COURSE)

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	TH	Pr/Or	Tut	Total
1	MI1104301	**Sem III** Introduction to Block chain	3	2	-	60	40	25	25	-	150	3	1	-	4
2	MI1104401	**Sem IV** Decentralize & Blockchain Technologies	3	2	-	60	40	25	-	25	150	3	1	-	4
3	MI1104501	**Sem V** Smart Contract & Cryptocurrency	3	2	-	60	40	25	-	25	150	3	1	-	4
4	MI1104601	**Sem VI** Blockchain Solutions	3	2	-	60	40	25	25	-	150	3	1	-	4
5	RP1104702	**Sem VII** Project	-	8	-	-	-	50	50	-	100	-	4	-	4
		Total	**12**	**16**	**-**	**240**	**160**	**150**	**100**	**50**	**700**	**12**	**8**	**-**	**20**

B.Tech
(Computer Science & Engineering)
Semester- VII

PEC-III Robotic Process Automation (RPA)

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03	University Examination	60		
Practical:	02	Internal Assessment:	40	Lecture :	03
		Term Work:	25	Practical:	01
		Practical:	25		
		Total:	150	Total:	04

Course Objective:

1. Understand the fundamentals, lifecycle, and business value of RPA and process automation.
2. Identify and document automation candidates using process discovery and design bestpractices.
3. Develop, test, and deploy robust RPA solutions with proper exception handling, logging, and governance.
4. Develop automations for web/desktop workflows including reliable UI interaction and strong exception handling, logging, and debugging practices.
5. Implement data handling and integrations: Excel/CSV/file/email automation, database connectivity, and REST API integration with secure credential handling and basic security practices.
6. Apply enterprise-grade RPA practices including orchestration, queues, retries, scheduling, monitoring, testing, governance/CoE principles, and document automation basics (PDF/OCR) for maintainable deployments.

Prerequisite:

Basic programming, fundamentals of databases, and basic understanding of web and Excel operations.

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain RPA concepts, bot types (attended/unattended), and the end-to-end RPA lifecycle.
2. Analyze business processes for automation suitability and prepare PDD/SDD documentation.
3. Build bots to automate web/desktop workflows and handle data in Excel/CSV, files, email, and PDFs.
4. Integrate RPA solutions with databases and REST APIs, including basic authentication and JSON parsing.
5. Implement enterprise features: queues, retries, scheduling/orchestration, versioning, and monitoring.
6. Apply testing, security, and governance practices to deliver maintainable, compliant automations.

Unit I **06 Hours**

RPA Overview: definition, use-cases, benefits and limitations. RPA vs macros, scripting, and AI automation. RPA lifecycle. Attended vs unattended automation. Process suitability and ROI. Tool ecosystem overview and studio fundamentals. Basic workflow constructs: variables, data types, sequences/flowcharts, control flow, logging basics.

Unit II **06 Hours**

Process Discovery, Documentation, and Design:

Process discovery and mapping. Automation feasibility: rule-based criteria, stability, exception rates, compliance needs. Documentation artifacts: Process Definition Document (PDD), Solution Design Document (SDD), test plan. UI automation concepts: selectors, screen scraping, waits, anchors. Reusability: modular workflows, libraries, config files. Debugging tools and troubleshooting strategies.

Unit III **06 Hours**

Web Automation and Data Handling:

Web automation: navigation, form fill, downloads/uploads, dynamic pages, synchronization (explicit/implicit waits). Desktop automation: common patterns for legacy apps. Data handling: Excel automation (read/write, formatting, pivot-like summaries), CSV/text files, regex and string parsing. Email automation: Outlook/Gmail integration patterns. File/folder automation: copy, move, rename, archive, and audit logs.

Unit IV **06 Hours**

Integration with Databases and APIs:

Database connectivity fundamentals: connection strings, secure credentials, executing queries, basic CRUD operations. Transaction processing design. REST APIs: HTTP methods, headers, tokens, JSON serialization/deserialization, pagination and error codes. Integrations with cloud storage and collaboration tools (conceptual). Security basics: credential vaults/assets, least privilege, secrets management.

Unit V **06 Hours**

Enterprise RPA: Orchestration, Queues, and Monitoring:

Orchestration concepts: robot provisioning, packages, environments, assets, schedules, triggers. Work queues: transaction items, retry policies, exception taxonomy (business vs system). Scalability and bot operations model. Monitoring and alerts, logs, audit trails, SLAs. Version control and release management. Introduction to CI/CD for RPA (conceptual).

Unit VI **06 Hours**

Document Automation, Testing, and Governance:

PDF and document automation: text extraction, templates, OCR basics, document understanding pipeline (classification, extraction, validation). Human-in-the-loop validation. Testing strategies: unit/integration/UAT, test data management, performance considerations. Governance and Center of Excellence concepts. Compliance, privacy, and secure handling of sensitive data. Capstone planning and evaluation rubric.

Textbooks

1. Leslie Willcocks, Mary C. Lacity, and Andrew Craig, "Robotic Process Automation and Risk Mitigation", SB Publishing.
2. Tripathi, A. M., Learning Robotic Process Automation: Create Software Robots and Automate Business Processes, Packt Publishing, 2018.
3. Murphy, C. M., Robotic Process Automation: Guide to Building Software Robots, Automate Repetitive Tasks & Become an RPA Consultant, Independently Published, 2019.

Reference Books

1. Mary C. Lacity and Leslie Willcocks, "Robotic Process Automation and Cognitive Automation", SB Publishing.
2. Syed, R., Robotic Process Automation using UiPath StudioX, BPB Publications, 2021.

List of Laboratory Exercises

1. Studio setup, first bot: file operations and logging.
2. Excel automation: cleaning, aggregation, report export.
3. Web automation: form fill, download and data extraction.
4. Email automation: send notifications with attachments.
5. Regex-based parsing and validation pipeline.
6. Database connectivity: query and update records securely.
7. REST API integration: GET/POST and JSON parsing and error handling.
8. Queue-based transaction processing with retries and exceptions.

9. Orchestrator deployment: package, schedule, monitor runs.
10. Document automation: PDF/OCR extraction with validation.

Project Based Learning

1. Develop an RPA bot to collect student application data from web forms, validate entries, and update a relational database sending automatic acknowledgment messages with proper logging.
2. Build a campus placement automation that reads student records, applies eligibility rules, and generates a shortlist report.
3. Create a cognitive finance bot to download fee receipts and extract key fields using visual OCR tools, updating the accounts ledger and sending a daily reconciliation summary report.
4. Design an enterprise invoice processing bot that extracts invoice details, validates against purchase order data, enters records into a portal, and manages the bot deployment pipeline.
5. Develop an IT helpdesk automation to read incoming tickets, categorize requests using generative language models, auto-respond for standard issues, and escalate complex cases.
6. Build an RPA bot that monitors a Carla simulator output directory, extracts logs containing variable camera frame rates, validates the telemetry data, and securely uploads the datasets to a cloud database for training visual language models or camera controllers.
7. Create a retail bot that scrapes competitor websites for specific product prices using Python scripts, applies statistical models to determine pricing trends, and automatically updates the company's frontend web catalog with competitive price adjustments.
8. Design a customer relationship management (CRM) bot that fetches daily customer reviews from social media APIs, classifies the sentiment of each review, and automatically routes negative feedback to the retention team's dashboard as high-priority tickets.
9. Build a cybersecurity audit bot that parses massive daily firewall and server access logs to identify unusual login anomalies or unauthorized IP threats, automatically generating and distributing a secure threat report to the network administration team.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

PEC-III Augmented and Virtual Reality

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03	University Examination:	60		
Practical:	02	Internal Assessment:	40	Lecture:	03
		Term Work:	25	Practical:	01
		Practical:	25		
		Total:	150	Total:	04

Course Objective:

1. To gain knowledge of historical and modern overviews and perspectives on virtual reality.
2. To Learn the fundamental Computer Vision, Computer Graphics Techniques related to VR/AR.
3. To familiarize students with AR/VR hardware components and tracking technologies.
4. To enable students to design and develop basic AR and VR applications using modern SDKs and game engines.
5. To understand interaction techniques, user experience (UX) design principles.
6. To expose students to advanced trends, research challenges, and real-world applications of AR/VR in various domains.

Prerequisite:

Basic Mathematics, Programming and Problem Solving, Engineering Graphics and Solid Modelling.

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand fundamentals of AR and VR systems.
2. Apply 3D graphics concepts in immersive environments.
3. Develop basic AR/VR applications using modern tools.
4. Analyze hardware and tracking technologies in AR/VR.
5. Design interactive virtual environments.
6. Evaluate emerging trends and future scope of AR/VR technologies.

Unit I **06 Hours**

Introduction to Augmented and Virtual Reality: Introduction to Augmented, Virtual and Mixed Reality, Evolution of VR and AR technologies, difference between AR,VR, and MR, Challenges and limitations of AR/VR systems, AR systems and functionality, Components of VR and AR systems, Applications of AR and VR.

Unit II **06 Hours**

3D Graphics, Visualization and Geometric Modelling for AR/VR: Basics of 3D graphics pipeline,3D Modeling concepts, the perspective projection, human vision, stereo perspective projection, Color theory, Conversion From 2D to 3D, 3D space curves, 3D boundary representation, Simple 3D modeling, 3D clipping, Illumination models, Reflection models, Shading algorithms, Geometrical Transformations: Introduction, Frames of reference, Modelling transformations.

Unit III **06 Hours**

Virtual Environment Input/Output Devices: Input (Tracker, Sensor, Digital Gloves, Movement Capture, Video based Input, 3D Menus & 3D Scanner, etc.), Output (Visual/Auditory/Haptic Devices) Generic VR system: Introduction, Virtual environment, Computer environment, VR technology, Model of interaction, VR Systems, Animating the Virtual Environment: Introduction, Linear and Nonlinear interpolation, the animation of objects, linear and non-linear translation.

Unit IV **06 Hours**

Augmented Reality Technologies and Development: AR fundamentals and working principles,

Marker-based and Markerless AR, Image recognition and object tracking in AR, AR SDKs and development tools, Mobile AR application development workflow, Designing AR user interfaces and usability principles, Case studies of AR applications.

Unit V **06 Hours**

Virtual Reality Technologies and Interaction: VR environment design and development, VR interaction techniques: Gesture-based interaction, Controller-based interaction, Voice interaction. Immersion and presence in VR, Navigation techniques in virtual environments, Haptic feedback systems,3D user interfaces (3D UI),VR content creation pipeline.

Unit VI **06 Hours**

Advanced Topics and Future Trends in AR/VR: Mixed Reality and Extended Reality (XR), AR/VR in Metaverse concepts, AI integration in AR/VR, Cloud-based AR/VR systems, WebXR and browser-based immersive technologies, Ethical issues and privacy concerns in AR/VR, Future trends and research directions in immersive technologies, Industrial case studies and real-world implementations.

Textbooks

1. Coiffet, P., Burdea, G. C., (2003), "Virtual Reality Technology," Wiley-IEEE Press, ISBN: 9780471360896.
2. Schmalstieg, D., Höllerer, T., (2016), "Augmented Reality: Principles & Practice," Pearson, ISBN: 9789332578494.
3. Hassanien, A. E., Gupta, D., Khanna, A., Slowik, A., (2022), "Virtual and Augmented Reality for Automobile Industry: Innovation Vision and Applications," Springer, ISBN: 9783030941017.

Reference Books

1. Steven M. LaValle, "Virtual Reality", Cambridge University Press, 2016.
2. Craig, A. B., (2013), "Understanding Augmented Reality, Concepts and Applications," Morgan Kaufmann, ISBN: 9780240824086.
3. Craig, A. B., Sherman, W. R., Will, J. D., (2009), "Developing Virtual Reality Applications, Foundations of Effective Design," Morgan Kaufmann, ISBN: 9780123749437.
4. John Vince, J., (2002), "Virtual Reality Systems, " Pearson, ISBN: 9788131708446

Anand, R., "Augmented and Virtual Reality," Khanna Publishing House.

t-Based Learning:

1. Modern Apartment Walk Simulation using Virtual Reality.
2. Farming in Village Simulation Using VR Tractor.
3. Large Warehouse Simulation with Forklift Virtual Reality.
4. Industry safety training simulation virtual reality.
5. Control and monitoring of IoT devices using mixed reality in unity engine using Arduino.
6. Factory machine simulation using virtual reality.
7. Augmented Reality in Education.
8. Product video promotion app using augmented reality.
9. AR in the automotive industry.
10. Graphics processor unit using virtual reality.
11. VR Virtual Tour of College Campus.
12. VR-Based Training Simulator (Lab/Industry).
13. AR Book Reader with 3D Model Visualization.
14. AR Museum Guide Application.
15. AR Solar System Educational Application.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

PEC-III Information Storage and Retrieval

Teaching Scheme		Examination Scheme		Credit Scheme	
Hours/Week			Marks		Credits
Lecture:	03	University Examination:	60		
Practical:	02	Internal Assessment:	40	Lecture:	03
		Term Work:	25	Practical:	01
		Practical:	25		
		Total:	150	Total:	04

Course Objective:

1. To Understand principles and architecture of Information Retrieval (IR) systems.
2. To Learn indexing, ranking, clustering, and search algorithms.
3. To Evaluate IR system performance, usability, and user interaction using standard metrics and methods.
4. To Explore web search, multimedia retrieval, and distributed information retrieval environments.
5. To Apply modern retrieval approaches including neural, semantic, and AI-driven search techniques.
6. To Examine recommender systems and intelligent retrieval while understanding ethical, responsible, and privacy-aware search technologies.

Prerequisite:

1. Data Structures & Algorithms
2. Basic Probability & Linear Algebra

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain IR architecture and text processing techniques.
2. Design scalable indexing and retrieval solutions.
3. Evaluate IR performance using standard and user-centric metrics.
4. Implement web, multimedia, and semantic retrieval systems.
5. Apply neural retrieval, vector search, and AI-based ranking.
6. Analyze bias, fairness, privacy, and ethical issues in retrieval systems.

Unit I **06 Hours**

Introduction to Information Retrieval: Basic Concepts of IR Information Retrieval vs Data Retrieval, IR system architecture &Pipeline, Text processing: tokenization, stemming, lemmatization, Index term weighting (TF-IDF, probabilistic weighting), Clustering & similarity measures, Neural text representations & word embeddings, Semantic similarity basics.

Unit II **06 Hours**

Indexing& Retrieval Models : Indexing Techniques: Inverted index & positional index, Index compression & hashing, **IR Models:** Basic concepts, Boolean Model, Vector Model, Probabilistic Model, Language models for IR, Learning to Rank, Neural ranking models (BERT-based retrieval), Dense retrieval & vector similarity search.

Unit III **06 Hours**

Query Processing & Intelligent Search: Query formulation & query expansion, Relevance feedback & pseudo relevance feedback,Query optimization, Pattern & structural queries, Fuzzy search & approximate matching, Cross-language information retrieval, Conversational search & voice queries, Natural language question answering systems, Context-aware & intent-aware search.

Unit IV **06 Hours**

Evaluation, UX & Human-Centered Retrieval: Performance Evaluation: Precision, Recall, F-measure, Mean Reciprocal Rank (MRR), NDCG& MAP, User-centric evaluation **IR Interface & Visualization:** Search interface & UX design, Behavioral analytics & implicit feedback, Personalization & user modeling, Trust& transparency in search results.

Unit V **06 Hours**

Web Search, Mining & Privacy-Aware Retrieval: Search Engine Architecture & Crawling, Link Analysis (PageRank, HITS), Web mining & SEO signals, Spam Detection & misinformation retrieval, Privacy-aware search & anonymization, Fair ranking & algorithmic transparency.

Unit VI **06 Hours**

Advanced & Emerging Retrieval Systems: Distributed & Multimedia Retrieval: Federated search & source selection, Multimedia retrieval (image, audio, video basics) **AI & Semantic Retrieval:** Knowledge graphs & semantic search, Vector databases & dense retrieval, Retrieval-Augmented Generation (RAG), AI-powered search systems **Recommender Systems:** Collaborative & hybrid filtering, Context-aware recommendation basics **Ethics & Responsible AI in IR:** Bias & fairness in search and ranking, Filter bubbles & algorithmic transparency, Data privacy, user consent & responsible personalization.

Textbooks:

1. Ricardo Baeza-Yates & Berthier Ribeiro-Neto — Modern Information Retrieval.
2. Christopher Manning, Raghavan & Schütze — Introduction to Information Retrieval.
3. Hang Li — Learning to Rank for Information Retrieval.

Reference Books:

1. David Grossman & Ophir Frieder — IR Algorithms & Heuristics.
2. Ricci et al. — Recommender Systems Handbook.
3. Zhang Jin — Visualization for Information Retrieval.

Project Based Learning

1. Mini Search Engine for College Website.
2. Semantic Search Engine using Word Embeddings.
3. AI-powered Question Answering System.
4. Personalized News Recommendation System.
5. Movie Recommendation System Using Collaborative Filtering.
6. Voice-based Search Interface using NLP.
7. Image Similarity Search System.
8. Fake News Detection & Retrieval System.
9. Academic Paper Search & Ranking System.
10. Resume Retrieval & Candidate Ranking System.
11. Web Crawler & Topic-based Indexing System.
12. E-commerce Product Search with Ranking Optimization.
13. Plagiarism Detection using Text Similarity.
14. Cross-language Information Retrieval System.
15. Chatbot with Retrieval-Augmented Response System (RAG).

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Deep Learning

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03	University Examination:	60		
Practical:	02	Internal Assessment:	40	Lecture:	03
		Term Work:	25	Practical:	01
		Practical:	25		
		Total:	150	Total:	04

Course Objective:

1. To understand mathematical foundations of deep learning.
2. To design, train, and evaluate deep neural networks.
3. To apply CNNs and sequence models to real-world problems.
4. To understand attention mechanisms and transformer architectures.
5. To explore generative AI and modern deep learning trends
6. To implement and deploy deep learning models.

Prerequisite:

Knowledge of Programming (Python preferred), Linear Algebra, Probability & Statistics, Basic Machine Learning concepts

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain and derive backpropagation and optimization algorithms.
2. Design and implement deep neural networks.
3. Apply CNNs for computer vision tasks.
4. Implement sequence models and understand transformers.
5. Evaluate model performance using appropriate metrics.
6. Develop and deploy a deep learning solution for real-world problems.

Unit I **06 Hours**

Introduction To Deep Learning: Introduction to Deep Learning: Basics: Biological Neuron, Idea of computational units, History of Deep Learning McCulloch Pitts Neuron, Thresholding Logic, Perceptron's, Perceptron Learning Algorithm, Multilayer Perceptron's (MLPs), Representation Power of MLPs, Activation Functions: Sigmoid, Tanh, ReLU, Softmax, Loss Functions: MSE, Cross Entropy, Gradient Descent, Backpropagation, Feed forward Neural Networks, Representation Power of Feed forwardNeural Networks, Model Evaluation Metrics.

Unit II **06 Hours**

Optimization and Regularization:

Gradient Descent -Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam, Learning Rate Scheduling, Weight Initialization (Xavier, He Initialization), Bias-Variance Tradeoff.

Regularization Techniques: Bias Variance Tradeoff, L1/L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout, Batch Normalization, Hyperparameter Tuning (Grid/Random Search).

Unit III **06 Hours**

Convolutional Neural Networks(CNN):

Convolution Operation, Padding, Stride, Pooling, CNN Architecture Design, **Classic Architectures:**LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet, Width and Depth of Neural Networks, Transfer Learning, Data Augmentation Techniques, Object Detection(YOLO/R-CNN), **Image Segmentation:** Introduction to U-Net, Visualization of CNN, Evaluation for Image Model.

Unit IV **06 Hours**

Sequence Models and Transformers:

Recurrent Neural Networks, Backpropagation through time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT, GRU, LSTMs, Attention Mechanism,Transformer Architecture (High-Level), BERT & GPT Overview, Sequence-to-Sequence Models, Applications in NLP:Text Classification, Sentiment Analysis, Chatbots, Text Generation.

Unit V **06 Hours**

Advanced Training Strategies:

Challenges in Deep Network Training,Gradient Clipping,Transfer Learning: Fine-tuning vs Feature Extraction,Model Ensembling,Mixed Precision Training, Model Compression &Pruning,Computational Complexity in Deep Learning,Experimental Design & Performance Analysis.

Unit VI **06 Hours**

Generative AI and Modern Trends:

Autoencoders & Variational Autoencoders (VAE), Generative Adversarial Networks (GANs) – Basic Concept, Diffusion Models, Introduction to Large Language Models, Prompt Engineering Basics, Overview of Hugging Face Ecosystem, Deep Learning in:Healthcare, Autonomous Systems, Computer Vision, Speech & Audio, Ethical Issues in Deep Learning, AI Bias & Responsible AI.

Textbooks

1. Nikhil Buduma, “Fundamentals of Deep Learning: Designing Next-Generation MachineIntelligence Algorithms”, O’Reilly publications.
2. Francois Chollet "Deep Learning with Python", Manning Publications, 2017
3. Deep Learning: An MIT Press Book By Ian Goodfellow and Yoshua Bengio and Aaron Courville

Reference Books

1. Josh Patterson, Adam Gibson, “Deep Learning: A Practitioners Approach”,

O‟REILLY, SPD, ISBN: 978-93-5213-604-9, 2017 Edition 1st.

2. Michael Nielsen, Neural Networks and Deep Learning, Determination Press, 2015.
3. Giancarlo Zaccone, Md. RezaulKarim, Ahmed Menshawy "Deep Learning with TensorFlow:
4. Explore neural networks with Python", Packt Publisher, 2017.
5. Antonio Gulli, Sujit Pal "Deep Learning with Keras", Packt Publishers, 2017.

List of Laboratory Exercises

1. Implement a Multilayer Perceptron for a binary classification problem (e.g., XOR or simple dataset) using only NumPy.
2. Develop a regression (e.g., house price prediction) or classification (e.g., MNIST digit classification) model using TensorFlow or PyTorch.
3. Train a CNN model on CIFAR-10 or MNIST dataset and evaluate performance using accuracy and confusion matrix.
4. Use a pretrained ResNet or VGG model (ImageNet weights) and fine-tune it for a custom dataset.
5. Perform sentiment classification using IMDB or SMS spam dataset using LSTM.
6. Fine-tune a pretrained BERT model for sentiment classification using Hugging Face.
7. Build an autoencoder for image denoising or dimensionality reduction using MNIST dataset.
8. Experiment with learning rate, batch size, dropout rate, and compare results.
9. Train the same model using SGD, RMSProp, and Adam and compare:Accuracy, Loss curves, Convergence speed.
10. Create a simple web application that allows users to input data and get predictions from trained model.
11. Object detection using Convolution Neural Network.

12. Develop a neural collaborative filtering model using sales or movie dataset.
13. Use LSTM for sentiment classification and generate a word relationship network graph for visualization.

Project Based Learning

1. Pneumonia Detection from Chest X-Ray Images
2. Brain Tumor Classification using MRI
3. Diabetic Retinopathy Detection
4. Crop Disease Detection System
5. Smart Irrigation Recommendation System
6. Product Recommendation System
7. Sales Forecasting using LSTM
8. Customer Sentiment Analysis from Reviews
9. Driver Drowsiness Detection
10. Face Mask Detection System
11. Resume Screening System
12. Object Detection for Waste Segregation
13. AI-Based Medical Report Generator
14. Crop Yield Prediction using Satellite Data

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Blockchain Technology

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03	University Examination:	60		
Practical:	02	Internal Assessment:	40	Lecture	03
		Term Work	25	Practical	01
		Oral	25		
		Total	150	Total	04

Course Objective:

1. Understand role of blockchain in Web 3.0.
2. Understand the bitcoin blockchain platform and its terminologies.
3. Understand Ethereum architecture.
4. Study and developed smart contracts, DAPPS for different application.
5. Understand enterprise blockchain.
6. To understand challenges to apply blockchain in emerging areas.

Prerequisite:

1. Cyber Security, Network security, Distributed networks
2. Object Oriented programming language.

Course Outcomes:

1. Differentiate between Web 2.0 and Web 3.0 with respect to various applications.
2. Elaborate the bitcoin mining, DLT, Consensus algorithm.
3. Analyse the Ethereum architecture.
4. Analyse the Hyperledger Fabric architecture.
5. Design smart contract and DAPP for real time application.
6. Illustrate blockchain integration with emerging technologies and security issues.

Unit I Fundamentals of Blockchain **06 Hours**

Introduction to Blockchain , Why Blockchain Platform: Platform types, Public, Private, technology requirements forimplementation. Distributed Ledger, Introduction to cryptography-Encryption and Decryption-Ciphers,hashing algorithms-SHA-256 algorithm-Application of SHA algorithm, Web 2.0 and Web 3.0.

Unit II Bitcoin Blockchain **06 Hours**

Introduction to Bitcoin, Bitcoin Wallets, Bitcoin Block, Bitcoin Transaction, Bitcoin Network, Operation of Bitcoin Blockchain, Blockchain Architecture , Consensus mechanism: Proof of Work, Bitcoin (BTC) – Genesis Block, Buy Bitcoin, Transactions, Unspent Transaction Output (UTXO), Bitcoin Mining.

Unit III Ethereum Blockchain **06 Hours**

Introduction, Ethereum components: miner and mining node, Ethereum virtual machine, Ether, Gas, Transactions, accounts, swarm and whisper, Ethash, end-to-end transaction in Ethereum, architecture of Ethereum .

Unit IV Ethereum Smart Contracts **06 Hours**

Smart Contract, Smart Contract Lifecycle, Solidity, Solidity State and Variable Types, Solidity Functions, Solidity Compilation and Deployment, mapper function, ERC20 and ERC721 Tokens, comparison between ERC20 & ERC721, use cases of smart, contract, smart Contracts: Opportunities, Risks.

Unit V Enterprise Blockchain **06 Hours**

Introduction to Hyperledger, tools and frameworks, Hyperledger Fabric, Comparison between Hyperledger Fabric & Other Technologies, Distributed Ledgers. Hyperledger Fabric Architecture,

Componentsof Hyperledger Fabric: MSP, Chain Codes etc., Transaction Flow, Advantages of Hyperledger Fabric Blockchain, working of Hyperledger Fabric, Creating Hyperledger network, Case

Unit VI Blockchain integration and Research challenges **06 Hours**

Integrating Blockchain with cloud, IoT, AI, ERP, End to endblockchain integration, Risks and Limitations of Blockchain: Privacy& Security. Criminal Use of Payment Blockchains, The "Dark" Side of Blockchain.

List of Experiments:

1. Demonstration of Blockchain.
2. Installation of Ganache, Flask and Postman.
3. Demo of Remix-Ethereum IDE.
4. Write a Simple Smart Contract for Bank with withdraw and deposit functionality.
5. Working of Blockchain Transaction, DLT(https://andersbrownworth.com/blockchain/)
6. Implement program to convert given text in to hashes using SHA 256 algorithm.
7. Create simple wallet transaction from one account to another account using Metamask.
8. Connect Metamask to a Ganache Test Network.
9. Ether Transaction Using Ganache.
10. Write Hello World smart contract in a higher programming language (Solidity).
11. Write simple smart contract for User identity management using Solidity language.
12. Write simple smart contract for Crowd fund ERC20 token

    1. User creates a campaign.
    2. Users can pledge, transferring their token to a campaign.
    3. After the campaign ends, campaign creator can claim the funds if total amount pledged is more than the campaign goal.
    4. Otherwise, campaign did not reach its goal, users can withdraw their pledge.

13. Build NFT Application by writing smart contract

    English Auction

    English auction for NFT.

    Auction

    - Seller of NFT deploys this contract.
    - Auction lasts for 7 days.
    - Participants can bid by depositing ETH greater than the current highest bidder.
    - All bidders can withdraw their bid if it is not the current highest bid.

    After the auction

    - Highest bidder becomes the new owner of NFT.
    - The seller receives the highest bid of ETH.

Creating a Business Network using Hyperledger Fabric.

Project Based Learning - Provisional List of Projects

1. Smart Contract: Development of smart block-based contact for project development.
2. Crypto-wallet: Creating a Crypto wallet for handling cryptocurrency.
3. Cryptocurrency: ERC-20 tokens & creating own cryptocurrency using solidity for Ethereum.
4. Blockchain-based Lottery – Picking a Winner from various Blockchain Nodes taking part in a lotter.
5. Organic Food Traceability System using Blockchain.
6. Pharma Supply Chain System using Smart Contracts.
7. Evidence Protection System Using Blockchain Technology.
8. Blockchain Shipment Management Tracking System.
9. Identifying Fake Products Through A Barcode-Based Blockchain System.
10. Medical Report Management & Distribution System on Blockchain.
11. Real Estate Booking System using Smart Contracts.

12. Trusted Crowdfunding Platform Using a Smart Contract.
13. Blockchain based Loan Management System with Smart Contracts.
14. Blockchain Rental Property System with Smart Contracts.
15. Anti-Money Laundering System using Blockchain.
16. Blockchain-based Transaction and Settlement System.

Textbooks

1. "Mastering Bitcoin, PROGRAMMING THE OPEN BLOCKCHAIN", 2nd Edition by Andreas M. Antonopoulos.
2. William Stallings, "Cryptography and Network security Principles and Practices", Pearson/PHI,2017.
3. Mastering Ethereum, Building Smart Contract and Dapps, Andreas M. Antonopoulos Dr. Gavin Wood, O'reilly.

Reference Books

1. "Mastering Blockchain", by Imran Bashir, Third Edition, Packt Publishing.
2. Blockchain with Hyperledger Fabric, LucDesrosiers, Nitin Gaur, Salman A. Baset, Venkatraman Ramakrishna, Packt Publishing.
3. Atul Kahate, Cryptography and Network Security, Tata Mc Grawhill, India, 2019.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

MAJOR PROJECT

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	-	University Examination:	-		
Practical:	16	Internal Assessment:	-	Theory:	-
		TermWork:	100	Practical:	08
		Practical:	50		
		Total:	150	Total:	08

Pre-requisites:
Basics of Software engineering, Software testing and knowledge of core computer engineering subjects.

Course Objectives:

1. To develop problem solving abilities using mathematics.
2. To apply algorithmic strategies while solving problems.
3. To develop time and space efficient algorithms.
4. To develop software engineering documents and testing plans.
5. To use algorithmic solutions using distributed, Embedded, concurrent and parallel environments.

Course Outcomes: On completion of the course, students will have the ability to:

1. Review and understand how previous experiences had an impact on affective states and intellectual performance.
2. Identify and define the problem.
3. Decide critically to solve the problem.
4. Demonstrate the ability to synthesize complex information from a variety of sources in decision-making.
5. Predict and develop a group process and desired outcomes.
6. Plan and perform collaboratively towards a common purpose.

Guidelines for Students:

1. The project will be undertaken preferably by a group of at least 3- 4 students who will jointly work and implement the project over the academic year. The work will involve the design of a system or subsystem in the area of Computer Engineering.
2. If the project is chosen a hardware project it will involve the designing a system or subsystem or upgrading an existing system. The design must be implemented into a working model with necessary software interfacing and a user manual.
3. If the project is chosen in the pure Software Application it must involve the detail Software Design Specifications, Data Structure Layout, File Design, Testing with complete documentation and user interface, with life cycle testing and as an executable package.
4. The group will select a project with the approval of the guide (Staff members assigned) and student submit the name of the project with a synopsis of 2 or 3 pages in the month of August in the academic year. A preliminary study report by the group must be submitted and certified at the end of seventh Semester.
5. It is expected that at least one research paper is published by each group with guide.

The project report stage-I will contain the details.
Problem definition and requirement specification, acceptance test procedure (ATP).

a) System definition, requirement analysis.
b) System design with UML.
c) Documentation and references.

Documentation will use UML approach with Presentation, Category, Use Case, Class Diagrams, etc

B.Tech
(Computer Science & Engineering)
Semester- VIII

INTERNSHIP

Teaching Scheme		Examination Scheme		Credit Scheme	
Hours/Week			Marks		Credits
Lecture:	-	University Examination :	-		
Practical:	-	Internal Assessment:	-	Lecture:	-
		Oral:	100	Oral:	12
		TermWork:	100		
		Total:	200	Total:	12

Pre-requisites:

Professional Skills, Knowledge of core computer engineering subjects.

Course Objectives:

1. To provide exposure for the students on practical engineering fields
2. To have better understanding of engineering practice in general and a sense of frequent possible problems.
3. To develop problem Identification abilities in real world
4. To experience use of technology /tools for software development

5. To identify their skills, values, beliefs, interests and personal abilities to develop the skills.
6. To prepare and present a report.

Course Outcomes: On completion of the course, students will have the ability to:

1. Propose a solution to solve real world problems with the help of technology.
2. Apply software engineering principles.
3. Evaluate and compare the various methodologies to solve a real-world problem.
4. Report hands on experience of using modern software development tools.
5. Assess their skills, values, beliefs, interests and personal abilities and act in congruence with them.
6. Identify social and ethical responsibilities and develop skills to compete for lifelong learning

As a part of the B.Tech Computer Science Engineering curriculum, Industrial Training is a Practical course, which the students B. Tech Computer Science Engineering should undergo in reputed Private / Public Sector / Government organization / companies as industrial training of 60 days weeks to be undergone by the student in the summer vacation after the semester VI. Examination and Oral examination will be conducted at the end of the semester VI.

The Industrial Training Report:

An Industrial Training report should be prepared by each student. The report is expected to demonstrate development of practical and professional skills in Engineering through technical experience and application of theoretical knowledge. Development of skills in dealing with people, and communication skills form part of the training experience. Students should seek advice from their employers to ensure that no confidential material is included into the report.The student should be able to present the report to prospective employers,

The following should be observed:

- Length of training
- Preliminary information
- Technical report/diary References should be made in the text to books, technical papers, standards etc., used during the training period and should be listed.
- Finally, a conclusion should include comprehensive comments on the type and value of experience gained, and how this relates to your professional career.
- A copy of the report should be submitted to his/her employer, another copy to the Department (through the respective Adviser).
- Students should also retain a personal copy of the report.

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03	University Examination:	60		
Practical:	-	Internal Assessment:	40	Lecture:	03
		Term Work:	-	Practical:	-
		Practical:	-		
		Total:	100	Total:	03

Course Objective:

1. To understand the fundamentals of image formation, representation, and the basic computer vision pipeline.
2. To apply image processing techniques such as filtering, edge detection, and morphological operations for feature extraction.
3. To design and implement object detection and recognition systems using classical methods and deep learning approaches (CNN, YOLO, etc.).
4. To design object detection and segmentation systems using CNNs, YOLO and U-Net.
5. To analyze camera calibration, stereo depth estimation, optical flow and basic 3D reconstruction
6. To evaluate GANs, Vision Transformers, and transfer learning for real-world CV applications.

Prerequisite:
Digital Image Processing, Linear Algebra, Python Programming, Basic Machine Learning concepts.

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain the principles of image formation, digitization, and the computer vision processing pipeline.
2. Apply spatial filtering, edge detection, morphological operations, and histogram equalization for image pre-processing.
3. Implement feature detectors (Harris, FAST) and descriptors (SIFT, ORB, HOG) for image matching and object recognition tasks.
4. Design object detection and segmentation systems using CNNs, YOLO, Faster R-CNN and U-Net architectures.
5. Analyze camera calibration, stereo depth estimation, optical flow, and basic 3D reconstruction using SfM.
6. Evaluate GANs and Vision Transformers, apply transfer learning, and implement CV solutions for real-world applications.

Unit I : Introduction to Computer Vision **06 Hours**

Overview of CV: Applications, Challenges, **Image Formation:** Pinhole Camera, Perspective Projection, **Digital Image Representation:** Pixels, RGB, HSV, Grayscale, Image Acquisition and Digitization,**CV Pipeline:** Preprocessing, Feature Extraction, Recognition.

Unit II : Image Processing Fundamentals **06 Hours**

Spatial Domain Filtering: Convolution, Mean, Median Filters, **Edge Detection:** Sobel, Prewitt, Canny, LoG, Histogram Equalization, **Morphological Operations:** Dilation, Erosion, Opening, Closing, Noise Reduction Techniques.

Unit III : Feature Extraction and Description **06 Hours**

Point Feature Detectors: Harris, FAST, SIFT: Keypoint Detection and Descriptor, ORB: Fast Feature Descriptor, **Texture Features:** LBP, HOG for Object Detection, Feature Matching using RANSAC.

Unit IV : Object Detection and Recognition **06 Hours**

Classical Object Detection: Viola-Jones Haar Cascades, **ML for Vision:** SVM, k-NN Classifiers,

CNNs: Architecture, Training, Inference, **Region-Based CNNs**: R-CNN, Fast R-CNN, Faster R-CNN, **Single-Shot Detectors:** YOLO (v5, v8), **Semantic Segmentation:** U-Net, **Instance Segmentation:** Mask R-CNN.

Unit V : 3D Vision and Camera Geometry **06 Hours**

Camera Calibration: Intrinsic and Extrinsic Parameters, **Stereo Vision:** Epipolar Geometry, Depth Estimation, , **Optical Flow:** Lucas-Kanade Method, Structure from Motion,**3D Point Cloud:** Basics and Visualization.

Unit VI : Advanced Topics and Applications **06 Hours**

Image Generation and Synthesis: GANs, **Vision Transformers (ViT):** Self-Attention Mechanism, Transfer Learning and Fine-Tuning (ResNet, EfficientNet), **Video Understanding:** Action Recognition, **Real-World Applications:** Medical Imaging, Autonomous Vehicles,Ethics, Bias, and Fairness in Computer Vision Systems.

Textbooks

1. Szeliski, R. - Computer Vision: Algorithms and Applications, 2nd Ed., Springer, 2022.

Reference Books

1. Gonzalez, R. & Woods, R. - Digital Image Processing, 4th Ed., Pearson. | Goodfellow, I. et al. - Deep Learning, MIT Press. | Forsyth & Ponce - Computer Vision: A Modern Approach, Pearson.
2. Forsyth, D. & Ponce, J. – *Computer Vision: A Modern Approach*, Pearson.
   Shanmugamani, R. – *Deep Learning for Computer Vision*, Packt Publishing, 2018.
3. Hartley, R. & Zisserman, A. – *Multiple View Geometry in Computer Vision*, 2nd Ed., Cambridge University Press, 2004.

List of Practical Assignments

1. Apply Sobel, Canny edge detectors and morphological operations on given images and compare results.
2. Implement SIFT and ORB feature extraction and perform image matching between two images, compare speed and accuracy.
3. Build an SVM classifier using HOG features for at least 3 categories and evaluate using a confusion matrix.
4. Fine-tune a pretrained CNN (MobileNet or ResNet-50) on a small custom dataset using transfer learning, report accuracy and loss curves.
5. Implement object detection using YOLOv8 on a standard dataset, and visualize detection results with bounding boxes.
6. Perform camera calibration using a checkerboard pattern and compute intrinsic parameters and distortion coefficients.
7. Compute optical flow using Lucas-Kanade method and apply to motion detection in a video stream.
8. Compare Vision Transformer (ViT) and CNN (ResNet) on the same image classification dataset, and analyze accuracy and inference time.

Project Based Learning

1. Real-Time Object Detection System: Build an object detection application using YOLOv8 and OpenCV that detects and classifies objects from a live webcam or video feed with bounding boxes and confidence scores.
2. Image Classification using Transfer Learning: Fine-tune a pretrained CNN (MobileNet or EfficientNet) on a domain-specific dataset, achieve at least 85% accuracy, and deploy as a simple web application.
3. Automated Attendance System using Face Recognition: Build a face recognition system using OpenCV that detects and recognizes faces from a webcam, marks attendance with timestamps, and generates a simple report.

4. Parking Slot Detection System: Detect and count available vs. occupied parking slots from an overhead camera image using image processing and display the result on screen.
5. Medical Image Segmentation: Train a U-Net model to segment regions of interest from medical images (brain MRI or chest X-ray) and evaluate using Dice coefficient and IoU.
6. Sign Language Recognition System: Design a real-time hand gesture recognition system using MediaPipe and CNN/LSTM to classify Indian Sign Language (ISL) alphabets from webcam input, and display the recognized text on screen to assist communication for differently-abled users.
7. 3D Reconstruction using Structure from Motion: Reconstruct a simple 3D point cloud from multiple images of an object using OpenCV for feature matching and camera pose estimation, and visualize using Open3D.
8. GAN-based Image Synthesis: Train a GAN or use a pretrained diffusion model to generate synthetic images for a chosen domain, evaluate image quality using FID score, and demonstrate practical use for data augmentation.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

IT Project Management(DevOps)

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03	University Examination:	60		
Practical:	-	Internal Assessment:	40	Lecture:	03
		Term Work:	-	Practical:	-
		Practical:	-		
		Total:	100	Total:	03

Course Objective:

1. Explain the business and technical drivers for adopting DevOps.
2. Describe the evolution of Agile, Lean, and Cloud practices.
3. Design and implement CI/CD pipelines.
4. Apply automated testing and deployment strategies.
5. Analyze system reliability using monitoring metrics.
6. To explore the emerging tools used in the DevOps lifecycle.

Prerequisite: Software Engineering and Project Management, Cloud Computing

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain core DevOps principles
2. Analyze relationships between DevOps and Agile/Cloud
3. Design CI/CD workflows
4. valuate deployment and testing strategies
5. Analyze monitoring data for reliability
6. Implement DevOps tools in real scenarios

Unit I Introduction to DevOps **06 Hours**

What is DevOps? Role of DevOps Engineer, Introduction to Continuous Integration and Continuous Delivery Policies, DevOps Culture: Dilution of barriers in IT departments, Process automation, Agile Practices, Reason for adopting DevOps,
What and Who Are Involved in DevOps? Introduction to DevOps pipeline phases , Defining the Development Pipeline, Centralizing the Building Server, Monitoring Best Practices,

Unit II Microservices Architecture and Cloud Native Development **06 Hours**

Monolithic applications, Introduction to microservice architecture, Implementing a microservices Architecture, Pros and Cons of a microservice Architecture, Characteristics of microservice architecture, Monolithic applications and microservices compared, microservices best practices, Deployment strategies, Introduction to cloud computing, cloud computing deployment models, service models, why to use cloud, Principle of container based application design, Introduction to Docker, Serverless computing, orchestration,
Case study : Netflix: Migration from monolithic architecture to microservices to achieve scalability and resilience.

Unit III Continuous Integration and Test-Driven Development **06 Hours**

Introduction to continuous integration, time to market and quality, Build in a Continuous Integration Scenario, Code Repository Server, Continuous Integration Server, Introduction to Continuous Delivery and chain, Differentiate Continuous Integration and Continuous Delivery, Strategies for Continuous Delivery, Benefits of Continuous Integration and Continuous Delivery, Designing a CI and CD System, Building Continuous Integration and Continuous Delivery Pipelines, Continuous Database Integration, Preparing the Build for Release, Identifying the Code in the Repository, Creating Build Reports, Putting the Build in a Shared Location, Releasing the Build
Case study : Facebook: Use of continuous integration to enable rapid code integration and frequent

releases with automated testing.

Unit IV Continuous Deployment and Orchestration **06 Hours**

Implementing a testing Strategy: Types of Tests, Integration testing, managing defect backlogs, what is Continuous Deployment? Changes moving through the deployment pipeline, Trade-offs in the deployment pipeline, Basic Deployment pipeline, Deployment pipeline practices & Commit stage, Automated Acceptance Test Gate, Subsequent test stages, preparing to release, Implementing a deployment pipeline

Case study :Amazon: Implementation of automated deployment pipelines enabling frequent and reliable production releases

Unit V Continuous Monitoring and Site Reliability **06 Hours**

What is a monitoring system? Factors involved in monitoring systems, why monitoring is important, white-box and black-box monitoring, building a monitoring system, monitoring infrastructure and applications, collecting data, logging, creating dashboard, behavior driven monitoring, what is site reliability engineering? SRE and DevOps, roles, and responsibilities of SRE, common tools used by SREs

Case study :Google: Adoption of Site Reliability Engineering practices to maintain high availability and performance at scale.

Unit VI DevOps Tooling and Case Studies **06 Hours**

Continuous Development/ Version Control: Git, Serverless orchestration: Kubernetes, Container Technology: Docker, Continuous Integration: Jenkins, Continuous delivery: Jenkins, Continuous Deployment: Ansible, Continuous Testing: Selenium, Monitoring: Prometheus, Bug tracking tool: Jira, elk stack. Case study: Spotify: Using Docker, Bank of New Zealand, EtSy.

Textbooks

1. PierluigiRiti, “Pro DevOps with Google Cloud Platform”, Apress, ISBN: 978-1-4842-3896-7.
2. Katrina Clokie, “A Practical Guide to Testing in DevOps”, Lean Publishing published on 2017-08-01.
3. Jez Humble and David Farley, “Continuous Delivery”, Pearson Education, Inc, ISBN: 978–0–321–60191–9.

Reference Books

1. Viktor Farcic, “The DevOps 2.0 Toolkit: Automating the Continuous Deployment Pipeline with Containerized Microservices” .
2. Jennifer Davis and Katherine Daniels, “Effective DevOps: Building a Culture of Collaboration, Anity, and Tooling at Scale”, O’Reilly Media, Inc., ISBN: 978-1-491-92630.
3. Sanjeev Sharma and Bernie Coyne, “DevOps for Dummies”, John Wiley & Sons, Inc., 2nd IBM Limited Edition, ISBN: 978-1-119-04705-6.

List of Laboratory Exercises

1. Install and configure Git repository and branching workflow using Git.
2. Build and containerize a sample application using Docker.
3. Create a CI pipeline using Jenkins.
4. Write automated unit tests and integrate into pipeline (JUnit / PyTest).
5. Deploy containerized application using Kubernetes (Docker Desktop Kubernetes or Minikube on Windows).
6. Set up monitoring using Prometheus and Grafana.
7. Implement Blue-Green deployment simulation (Docker/Kubernetes).
8. Build complete CI/CD pipeline mini project.

Project Based Learning

1. CI pipeline for a simple web application using Jenkins.
2. Git branching and release workflow implementation using Git.
3. Containerizing a Node/Python app using Docker.
4. Kubernetes deployment of a containerized application using Kubernetes.
5. Automated configuration management using Ansible.
6. Infrastructure provisioning using Terraform.
7. Automated testing integration using Selenium.
8. Monitoring dashboard setup using Prometheus and Grafana.
9. Log monitoring using Elasticsearch and Kibana.
10. Simple CD pipeline using GitHub Actions.
11. Blue-Green deployment simulation using Docker.
12. DevSecOps pipeline with vulnerability scanning using OWASP ZAP.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI