NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA, GAUTAM BUDDH NAGAR

(AN AUTONOMOUS INSTITUTE)

Estd.2001
Get Future Ready

NIET
Greater Noida
An Autonomous Institute

Affiliated to

DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW

Evaluation Scheme & Syllabus

For

Bachelor of Technology

Computer Science & Engineering (Data Science)

Third Year

(Effective from the Session: 2026-27)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

CSE (Data Science)
Evaluation Scheme
SEMESTER V

Sl. No.	Subject Codes	Subject Name	Types of Subjects	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
WEEKS COMPULSORY INDUCTION PROGRAM														
1	BCSE0502A	Computer Networks	Mandatory	3	1	0	30	20	50		100		150	4
2	BCSDS0501	DevOps and Cloud Computing	Mandatory	3	1	0	30	20	50		100		150	4
3		Departmental Elective-II	Dept Elective	3	0	0	30	20	50		100		150	3
4		Departmental Elective-III	Dept Elective	3	0	0	30	20	50		100		150	3
5	BCSE0552A	Computer Networks Lab	Mandatory	0	0	4				50		50	100	2
6	BCSDS0551	DevOps and Cloud Computing Lab	Mandatory	0	0	2				25		25	50	1
7		Departmental Elective-II Lab	Dept Elective	0	0	2				25		25	50	1
8	BCSML0555B	Deep Learning for Data Analytics	Mandatory	0	0	6				50		100	150	3
9	BCSCC0501Y	Design Thinking-II	Mandatory	2	0	0	60	40	100				100	2
10	BCSCC0552	Research Methodology, Ethics and Writing	Mandatory	0	0	2				50			50	1
11	BEC0559X	Internship-II	Mandatory	0	0	2				50			50	1
12	BVAC0551	Professional Practice and Employability-I	Compulsory Audit	0	0	6	60	40	100					NA
		MOOCs (For B.Tech. Hons. Degree)	*MOOC's											
		TOTAL		**14**	**2**	**24**	**180**	**120**	**300**	**250**	**400**	**200**	**1150**	**25**

*** List of MOOCs (Infosys Springboard) Based Recommended Courses for 3rd year B. Tech Students**

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	BMC0011	Building Machine Learning Systems with TensorFlow	Infosys Wingspan (Infosys Springboard)	27h 18m	2
2.	BMC0095	ReactJS	Infosys Wingspan (Infosys Springboard)	61h 2m	4
3.	BMC0119	CCNA: Introduction to Networks	CISCO	70h	4

PLEASE NOTE: -

- **A 3-4 weeks Internship shall be conducted during summer break after semester-IV and will be assessed during semester-V.**
- **Compulsory Audit (CA) Courses (Non-Credit - BVAC0551)**
  - All Compulsory Audit Courses (a qualifying exam) do not require any credit.
  - The Total and obtained marks are not added in the Grand Total.

Abbreviation Used:

L: Lecture, T: Tutorial, P: Practical, CT: Class Test, TA: Teacher Assessment, PS: Practical Sessional, TE: Theory End Semester Exam. CE: Core Elective, OE: Open Elective, DE: Departmental Elective, PE: Practical End Semester Exam, CA: Compulsory Audit, MOOCs: Massive Open Online Courses.

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

CSE (Data Science)
Evaluation Scheme
SEMESTER VI

Sl. No	Subject Codes	Subject Name	Type of Subject	Periods			Evaluation Schemes				End Semester		Total	Credit
				L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	BCSDS0601	Big Data Analytics	Mandatory	3	1	0	30	20	50		100		150	4
2	BCSE0603	Software Engineering and Design	Mandatory	3	1	0	30	20	50		100		150	4
3		Departmental Elective-IV	Elective	3	0	0	30	20	50		100		150	3
4		Departmental Elective-V	Elective	3	0	0	30	20	50		100		150	3
5		Open Elective- I	Open Elective	2	0	0	30	20	50		50		100	2
6	BCSDS0653	Big Data Analytics Lab	Mandatory	0	0	4				50		50	100	2
7	BCSE0653Y	Software Engineering and Design Lab	Mandatory	0	0	4				50		50	100	2
8		Departmental Elective Lab	Elective	0	0	2				25		25	50	1
9	BCSE0654A	Computer Vision lab	Mandatory	0	0	6				50		100	150	3
10	BCSE0659X	Applied Research with Mini Project	Mandatory	0	0	2				50			50	1
11	BVAC0651	Professional Practice & Employability - II	Compulsory Audit	0	0	6	60	40	100					NA
12		MOOCs (For B.Tech. Hons. Degree)	*MOOC's											
		TOTAL		**14**	**2**	**24**	**150**	**100**	**250**	**225**	**450**	**225**	**1150**	**25**

*** List of MOOCs (Infosys Springboard) Based Recommended Courses for 3rd year B. Tech Students**

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1	BMC0074	Data Analysis with Pandas and Python	Infosys Wingspan (Infosys Springboard)	19h 49m	1.5
2	BMC0069	Big Data - 201	Infosys Wingspan (Infosys Springboard)	24h 13m	2
3	BMC0131	Python for Data Science	NASSCOM	30h	2

PLEASE NOTE: -

- **A 3-4 weeks Internship shall be conducted during summer break after semester-VI and will be assessed during Semester-VII**
- **Compulsory Audit (CA) Courses (Non-Credit - BVAC0651)**
  - All Compulsory Audit Courses (a qualifying exam) do not require any credit.
  - The Total and obtained marks are not added in the Grand Total.

Abbreviation Used:

L: Lecture, T: Tutorial, P: Practical, CT: Class Test, TA: Teacher Assessment, PS: Practical Sessional, TE: Theory End Semester Exam.
CE: Core Elective, OE: Open Elective, DE: Departmental Elective, PE: Practical End Semester Exam, CA: Compulsory Audit,
MOOCs: Massive Open Online Courses.

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

List of Department Elective (if any): -

S. No.	Subject code	Subject Name	Type of Subject	Bucket Name	Branch	Sem.
1	BCSAI0512	Advanced Java Programming	DE-2	Full Stack	DS	V
2	BCSAI0512P	Advanced Java Programming Lab	DE-2 Lab	Full Stack	DS	V
3	BCSE0514	Design Pattern	DE-3	Full Stack	DS	V
4	BCSDS0516	Predictive Analytics	DE-2	Data Analytics	DS	V
5	BCSDS0516P	Predictive Analytics Lab	DE-2 Lab	Data Analytics	DS	V
6	BCSDS0513	Product Analytics	DE-3	Data Analytics	DS	V
7	BCSDS0611	Applied Data Science and AI	DE-4	Data Analytics	DS	VI
8	BCSDS0611P	Applied Data Science and AI Lab	DE-4 Lab	Data Analytics	DS	VI
9	BCSDS0612	Natural Language Processing and Social Media Analytics	DE-5	Data Analytics	DS	VI
10	BCSE0614	Web Development Using Mean Stack	DE-4	Full Stack	DS	VI
11	BCSE0614P	Web Development Using Mean Stack Lab	DE-4 Lab	Full Stack	DS	VI
12	BCSE0615	Web Development Using Mern Stack with Devops	DE-5	Full Stack	DS	VI

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA
(An Autonomous Institute)

AICTE Guidelines in Model Curriculum:

A student will be eligible to get Under Graduate degree with Honours only, if he/she completes the additional MOOCs courses such as Infosys Springboard/NASSCOM/CISCO (as applicable) certifications, or any other online courses recommended by the Institute (Equivalent to 20 credits).
During Complete B.Tech. Program Guidelines for credit calculations are as follows.

1. For 6 to 12 Hours =0.5 Credit
2. For 13 to18 =1 Credit
3. For 19 to 24 =1.5 Credit
4. For 25 to 30 =2 Credit
5. For 31 to 35 =2.5 Credit
6. For 36 to 41 =3 Credit
7. For 42 to 47 =3.5 Credit
8. For 48 and above =4 Credit

For registration to MOOCs Courses, the students shall follow Infosys Springboard/NASSCOM/CISCO (as applicable) registration details as per the assigned login and password by the Institute these courses may be cleared during the B. Tech degree program (as per the list provided). After successful completion of these MOOCs courses, the students shall provide their successful completion status/certificates to the Controller of Examination (COE) of the Institute through their coordinators/Mentors only.

The students shall be awarded Honours Degree as per following criterion.

i. If he / she secures 7.50 as above CGPA.
ii. Passed each subject of that degree program in the single attempt without any grace.
iii. Successful completion of MOOCs based 20 credits

SYLLABUS- SEMESTER V

Course Code: BCSE0502A	Course Name: Computer Networks	L	T	P	C
Course Offered in B. Tech: CSE/CS/IT /CSE(AI)/CSE(AIML)/ CSE(DS)/ CSE(CYS)/MCT /ECE		3	1	0	4

Pre-requisite: Knowledge of basic computing

Course Objectives:

The course aims to provide students with a comprehensive understanding of computer network fundamentals, protocol architectures, addressing and routing mechanisms, transport and application layer services, and network security concepts. It also enables students to analyze, design, configure, and evaluate network systems and protocols to support efficient and secure data communication in modern computer networks.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand fundamentals of computer networks, components, performance metrics, multiplexing, topologies, and compare OSI and TCP/IP models with layer-wise responsibilities.	K2
CO2	Analyze data link layer mechanisms, MAC protocols, switching, VLANs, trunking, DTP, VTP, and STP operation for loop prevention and port state transitions.	K4
CO3	Model IPv4 and IPv6 networks, addressing schemes and configure routing protocols, while evaluating routing metrics, administrative distance, and loop prevention mechanisms.	K4
CO4	Analyze TCP and UDP behavior, congestion control, traffic shaping, and their impact on QoS, including connection management, flow control, and port usage.	K4
CO5	Apply application-layer and security solutions using core network services, NAT/PAT, cryptography, firewalls, ACLs, and threat mitigation techniques.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	-	-	-	-	-	1	-	3	2	3
CO2	2	2	2	2	3	-	-	-	1	1	1	2	2	2
CO3	3	3	3	1	2	-	-	-	1	1	1	1	2	1
CO4	2	2	1	1	2	-	-	1	1	1	1	2	1	3
CO5	2	2	2	1	2	1	1	2	1	1	1	2	2	3

Course Contents / Syllabus

Unit 1	**Introduction to Networks and Data Communication**	**08 hours**

Introduction to Computer Networks: Definition, evolution, goals, advantages, disadvantages, Client-server, peer-to-peer networks, modes of communications.
Types of networks: PAN, LAN, WAN, MAN.
Network components: End devices (PCs, servers, printers), Intermediary devices (gateways, routers, switches, APs), NICs.
Network media: Wired and Wireless media, along with performance metrics bandwidth, latency, throughput, and jitter.
Multiplexing Techniques: Frequency Division Multiplexing (FDM), Time Division Multiplexing (TDM).
Network representations: Topology diagrams including Bus, Ring, Star, Mesh, Hybrid, Physical vs. Logical Topologies.
Network Communication Model: ISO/OSI and TCP/IP protocol suite with responsibilities of each layer, comparison between models, PDU at each layer.

Unit 2	**Data Link Layer, Ethernet Switching, VLANs & STP**	**09 hours**

Data Link Layer: Framing, Error Detection and Correction, Flow control (Elementary Data Link Protocols, Sliding Window protocols).
Medium Access Control and Local Area Networks: Channel allocation, Multiple access protocols (ALOHA, Slotted ALOHA, CSMA/CD, CSMA/CA, Binary-backoff algorithm, LAN standards, Protocols and frame formats.
Switching and Techniques: Circuit, Message and Packet switching techniques.
Virtual Local Area Networks (LAN): VLAN concepts, VLAN types, VLAN configuration, 802.1Q trunking: VLAN tagging, native VLAN on trunk, Dynamic Trunking Protocol (DTP), VLAN Trunking Protocol (VTP), Static vs Dynamic VLANs.
STP (IEEE 802.1D): BPDU, root bridge election, port roles (Root, Designated, Non-Designated), port states (Blocking, Listening, Learning, Forwarding, Disabled).

Wireless Networking: WLAN standards, Wireless security standards, CSMA/CA: Hidden/exposed terminal problem, RTS/CTS mechanism, Wireless Security, Bluetooth.

Unit 3	**Network Layer: IP Addressing, Subnetting & Routing**	**09 hours**

Role of the Network Layer: Logical addressing, Path determination, Packet encapsulation.
IP Addressing: Need of IP address, Classification of IP addresses.
IPv4 Address: Notation, IPv4 classes, Public vs. Private address, Classes of IPv4 addresses, Loop back address, IPv4 Header format, ICMPv4.
Subnetting and VLSM: Need of subnetting, Subnet mask, Classless Inter-Domain Routing (CIDR), Variable Length Subnet Masking (VLSM).
IPv6 Addressing: Need of IPv6, difference between IPv4 and IPv6, unicast, multicast, anycast, IPv6 Header format, ICMPv6.
IPv4-to-IPv6 transition: Dual-stack, tunneling techniques (6-to-4, Intra-Site Automatic Tunnel Addressing Protocol (ISATAP), Teredo).
Routing Concepts: Router functions, Routing protocol metrics, Routing table structure, Split Horizon, Route Poisoning, Holddown Timers, Triggered Updates Distance vector vs Link state routing and its examples, Static Routing and its Configuration, Route types, Default route, Administrative Distance, Dynamic Routing Protocols, RIPv1, RIPv2, OSPF, EIGRP, BGP, IS-IS, Packet Forwarding methods.

Unit 4	**Transport Layer**	**10 hours**

Role of the Transport Layer: End-to-end communication, multiplexing/demultiplexing.
Transmission Control Protocol (TCP): Key Characteristics, packet format, flow control (window size), error recovery, Application areas,
Handshaking Techniques: TCP 3-Way Handshaking, four-way Teardown.
User Datagram Protocol (UDP): Key Characteristics, packet format, working, Application areas.
Port numbers: Well-known ports, Registered ports, Dynamic/Private ports.
TCP Socket: Life cycle, Socket programming basics, port scanning & enumeration, banner grabbing, socket-based attacks, man-in-the-middle attacks, packet sniffing, SSL/TLS, socket/session hijacking, firewall & socket filtering, intrusion detection systems, reverse shells & backdoors, Netcat usage, secure socket coding, encryption in socket communication, DoS/DDoS attacks.
Congestion Control Techniques: Open Loop and Closed Loop.
Traffic Shaping: Leaky Bucket Algorithm, Token Bucket Algorithm, Traffic shaping in Quality of Service (QoS).

Unit 5	**Application Layer and Security**	**09 hours**

Application Layer Protocols: DNS, DHCP, HTTP/HTTPS: request/response model, methods (GET, POST, PUT, DELETE), status codes, TLS/SSL, FTP (active/passive modes), SFTP, TFTP.
Email and its protocols: User Agent (UA), Message Transfer Agent (MTA), Message Access Agent (MAA), SMTP, POP3, IMAP.
Network management: SNMP (v1/v2c/v3).
NAT and PAT concepts: Static NAT and its configuration, Dynamic NAT and its configuration, PAT and its configuration
Security: Network threats: DoS/DDoS, reconnaissance, access attacks, MITM, social engineering, phishing, IPSec,
Introduction to Cryptography: Definition and goals of cryptography; Security services: Confidentiality, Integrity, Authentication, Non-repudiation; Types of attacks (passive vs active); Basic terminology: plaintext, ciphertext, key, encryption, decryption; Substitution techniques, Transposition techniques; Symmetric Key Cryptography: Concepts of symmetric encryption, Block ciphers and stream ciphers, Data Encryption Standard (DES); Asymmetric Key Cryptography: Public key cryptography concepts; RSA algorithm; Digital Signature.
Firewall: Types of Firewalls, Packet filtering concepts, Stateful vs stateless inspection, Firewall policies and rules; Access Control List (ACL): Purpose, packet filtering, traffic classification, QoS marking, Standard ACL, Extended ACL, Named ACLs, ACL wildcard masks, IPv6 ACLs: differences from IPv4 ACLs; implicit deny and permit ICMPv6.

	Total Lecture Hours	**45 hours**

Textbook

S NO.	Book Title	Author
1.	Data Communications and Networking, 4th Edition, McGraw-Hill Education	Behrouz A Forouzan.
2.	Computer Networking: A Top-Down Approach, 7th Edition, Pearson Education.	Hoboken
3.	Introduction to Networks Companion Guide (CCNAv7), 1st Edition, Cisco Press / Pearson Education, 2020.	Cisco Networking Academy
4.	Switching, Routing, and Wireless Essentials (SRWE), (CCNAv7), 1st Edition, Cisco Press / Pearson Education, 2020.	Cisco Networking Academy, Allan Johnson

Reference Books:

1. **Data and Computer Communications, Eighth Edition**

Self-Study Content Links:		Duration
MOOCs	1. CCNA: Introduction to Networks 2. CCNA: Switching, Routing, and Wireless Essentials (SRWE) Supplemental Unit 3. CCNA: Enterprise Networking, Security, and Automation (ENSA)	70 Hours each
Video Lectures	Complete CN Computer Networks in one shot | Semester Exam | Hindi https://www.youtube.com/watch?v=l_OPR2yh2co&list=PLh94XVT4dq02frQRRZBHzvj2hwuhzSByN&index=2 Lec-1: Computer Networks and Security Full Syllabus for GATE, UGC NET,DSSSB,NIELIT & University exam	1. 6Hrs 18Min 2. 11Hrs 54Min 3. 24 Hrs 58Min
Web References	https://www.geeksforgeeks.org/computer-networks/computer-network-tutorials/ https://www.geeksforgeeks.org/computer-networks/basics-computer-networking/ https://www.geeksforgeeks.org/computer-networks/ccna-tutorial-for-beginners/	-

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Course Code: BCDS0501	Course Name: DevOps and Cloud Computing	L	T	P	C
Course Offered in: CSE (Data Science)		3	1	0	4

Prerequisite: Basic knowledge of Operating Systems, Computer Networks, and Programming.

Course Objectives:
This course provides fundamental knowledge of cloud computing concepts, AWS services, DevOps lifecycle practices with Git/GitHub, build automation and CI/CD pipelines using Maven/Gradle/Jenkins, and containerization with Docker/Kubernetes and Ansible configuration management, enabling students to design complete DevOps workflows integrating cloud platforms, CI/CD automation, and monitoring tools.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand cloud computing concepts, service models, and core AWS services.	K2
CO2	Analyze DevOps principles, lifecycle, and version control using Git and GitHub.	K4
CO3	Apply build automation and CI/CD tools like Maven, Gradle, and Jenkins.	K3
CO4	Implement containerization and configuration management using Docker, Kubernetes, and Ansible.	K3
CO5	Design end-to-end DevOps pipelines with cloud, CI/CD, monitoring, and logging tools	K5

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	1	-	3	-	-	-	-	-	2	3	2	1
CO2	2	2	1	-	3	-	1	2	2	1	2	2	3	1
CO3	2	2	2	1	3	-	-	1	1	2	2	2	3	2
CO4	2	2	3	1	3	-	-	1	1	2	2	3	3	2
CO5	2	3	3	2	3	1	1	2	2	2	3	3	3	3

Course Contents / Syllabus

Unit 1	**Cloud Computing Fundamentals**	**10 hours**

Introduction: Introduction to Cloud Computing, Need and Advantages of Cloud Computing, Service Models: IaaS, PaaS, SaaS,
Deployment Models: Public, Private, Hybrid, Virtualization
Public Cloud Platforms: Overview of AWS, Microsoft Azure, Google Cloud
AWS Core Services: IAM (Users, Roles, Policies), EC2, S3, VPC, RDS
Modern Cloud Concepts: Serverless Computing (AWS Lambda), Infrastructure as Code (Terraform / CloudFormation)

Unit 2	**Decision Control and Looping Statements**	**10 hours**

DevOps: Definition, History, Principles, DevOps Lifecycle and Objectives, DevOps vs Agile vs Waterfall, DevOps on Cloud

Version Control Systems: Git Basics, 3-tree Architecture, Git Commands: Clone, Commit, Push, Pull, Fetch, Branching, Merging, Rebase, Stash, Reset

GitHub: Repository Management, Pull Requests & Code Reviews

CI/CD Introduction: Continuous Integration & Continuous Delivery, GitHub Actions (Basics)

DevSecOps Basics: Secure coding practices, Introduction to security in pipelines.

Unit 3	**Build Tools & Continuous Integration**	**8 hours**

Build Automation: Maven: Installation, POM, Lifecycle, Dependencies, Gradle: Basics and Build Process, Artifact Creation & Dependency Management

Continuous Integration: Jenkins Architecture & Setup, Jenkins Jobs & Pipelines, Pipeline as Code

Modern CI/CD Tools: GitHub Actions / GitLab CI (Overview), Artifact Repositories (Nexus / Artifactory)

Unit 4	**Containerization & Configuration Management**	**9 hours**

Docker: Introduction to Containers, Docker Architecture & Engine, Docker Images & Containers, Dockerfile & Commands, Docker Compose, Docker Hub

Container Orchestration: Kubernetes Basics, Architecture (Pods, Nodes, Clusters), Deployment & Scaling

Configuration Management: Ansible Architecture, YAML Basics, Inventory, Units, Playbooks & Roles

Unit 5	**DevOps Pipelines On Cloud**	**8 hours**

AWS DevOps Services: CodePipeline, CodeBuild, CodeDeploy

CI/CD Pipeline Implementation: End-to-End Pipeline (GitHub → Build → Docker → Deploy)

Monitoring & Logging: Prometheus & Grafana (Basics), ELK Stack Overview

	Total Lecture Hours	**45 hours**

Textbook:

1. **Thomas Erl , "Cloud Computing: Concepts, Technology & Architecture", Edition: 1st or later**
2. **Douglas E. Comer , "The Cloud Computing Book", Edition: 1st, 2023**
3. **Stephen Baron, "AWS: The Complete Beginner's Guide", Kindle Edition2020**
4. **Roberto Vormittag, "A Practical Guide to Git and GitHub for Windows Users: From Beginner to Expert in Easy Step-By-Step Exercises", Second Edition, Kindle Edition, 2016.**
5. **John Ferguson Smart , "Jenkins: The Definitive Guide", Edition: 2nd or later.**

Reference Books:

1. **Hands-On Azure Devops: Cicd Implementation For Mobile, Hybrid, And Web Applications Using Azure Devops And Microsoft Azure: CICD Implementation for ... DevOps and Microsoft Azure (English Edition) Paperback – 1 January 20202. By MiteshSoni**
2. **Jeff Geerling, "Ansible for DevOps: Server and configuration management for humans", First Edition, 2015.**
3. **David Johnson, "Ansible for DevOps: Everything You Need to Know to Use Ansible for DevOps", Second Edition, 2016.**
4. **MariotTsitoara, "Ansible 6. Beginning Git and GitHub: A Comprehensive Guide to Version Control, Project Management, and Teamwork for the New Developer", Second Edition, 2019.**

NPTEL/ YouTube/ Faculty Video Link:

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. Online Cloud Computing Training Course- Cloud Computing 101 (3h 56m) 2. AWS Cloud Practitioner Essentials_FSP (11h 05m) 3. DevOps Fundamentals (9h 10m)	24h 18m

Video Lectures	1. https://www.youtube.com/watch?v=F9i4_1u5Mzk 2. https://www.youtube.com/watch?v=k1RI5locZE4 3. https://www.youtube.com/watch?v=hQcFE0RD0cQ 4. https://www.youtube.com/watch?v=sz5gfkwPITE&list=PLhNrFKat_aeIogDQc0 xnEiZ2TLDKzZCEM	39 h
Web References	1. https://www.geeksforgeeks.org/cloud-computing/cloud-computing-tutorial/ 2. https://docs.aws.amazon.com/ec2/?icmpid=docs_homepage_featuredsvcs 3. https://www.tutorialspoint.com/amazon_web_services/index.htm 4. https://timesofcloud.com/aws/devops/	20 h
Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.		

Course Code: BCSDS0516	Course Name: Predictive Analytics	L	T	P	C
Course Offered in: CSE (Data Science)		3	0	0	3

Prerequisite: Basic Knowledge of R, Python and Statistics.

Course Objectives: This course enables students to solve complex problems by identifying hidden patterns in data, understand intricate relationships among interdependent variables, collect, analyze, and interpret large validated datasets, extract meaningful insights for data-driven decision-making, and apply analytical techniques for intelligent problem solving.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Apply specific statistical and regression analysis methods applicable to predictive analytics to identify new trends and patterns, uncover relationships, create forecasts, predict likelihoods, and test predictive hypotheses	K3
CO2	Learn how to select the appropriate method for predictive analysis, and how to build effective predictive models.	K4
CO3	Learn how to evaluate the soundness, appropriateness, and validity of their models and how to interpret and report on results for a management audience.	K5
CO4	Apply predictive analytics approaches on diverse business cases and scenarios	K3
CO5	Understand the process of formulating business objectives, data selection/collection, preparation, and process to successfully design, build, evaluate and implement predictive models for a various business application.	K6

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	2	-	2	-	1	-	2	2	3	2	2
CO2	3	3	2	2	2	2	1	1	1	1	2	2	2	3
CO3	3	3	2	3	2	2	2	1	1	2	2	3	2	3
CO4	3	2	3	2	2	1	2	1	1	1	2	2	2	2
CO5	3	3	2	2	2	2	1	-	1	2	2	3	2	2

Course Contents / Syllabus

Unit 1	LINEAR REGRESSION	9 hours

Introduction to Predictive Analytics, Regression basics: Relationship between attributes using Covariance and Correlation, Relationship, between multiple variables: Regression (Linear, Multivariate) in prediction, Residual Analysis Identifying significant features, feature reduction using AIC, multi-collinearity, Non-normality and Hetero scedasticity, Hypothesis testing of Regression Model, Confidence intervals of Slope, R-square and goodness of fit, Influential Observations – Leverage

Unit 2	MULTIPLE LINEAR REGRESSION	9 hours

Polynomial Regression, Regularization methods, Lasso, Ridge and Elastic nets, Categorical Variables in Regression.		
Unit 3	**NON-LINEAR REGRESSION**	**9 hours**
Logit function and interpretation, Types of error measures (ROCR), Logistic Regression in classification.		
Unit 4	**FORECASTING MODELS**	**9 hours**
Trend analysis, Cyclical and Seasonal analysis, smoothing, Moving averages, Box-Jenkins, Holt-winters, Autocorrelation, ARIMA, SARIMA Examples: Applications of Time Series in financial markets.		
Unit 5	**FEATURE ENGINEERING**	**9 hours**
Data Science Vs. Machine Learning, Exploratory Data Analysis, Feature Encoding, Imputation Feature Scaling, determining correlation, Feature selection, t-SNE, Feature extraction as an application based on Code Using (LDA, PCA & FA) as CASE STUDIES on a sample Data Set.		
	Total Lecture Hours	**45 hours**

Textbook:

1. An Introduction to Statistical Learning, Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, Springer, 2nd Edition, 2021.
2. Dean Abbott, "Applied Predictive Analytics Principles and Techniques for the Professional Data Analyst" by Wiley
3. Introduction to Linear Regression Analysis, Douglas C. Montgomery, Elizabeth A. Peck and G. Geoffrey Vining, Wiley, 5th Edition, 2012.
4. The Elements of Statistical Learning, Trevor Hastie, Robert Tibshirani and Jerome Friedman, Springer, 2nd Edition, 2009.

Reference Books:

1. Fundamentals of Machine Learning for Predictive Data Analytics by John D. Kelleher, Brian Mac Namee and Aoife D'Arcy.
2. Practical Regression and Anova using R, Julian J. Faraway, Lulu Press, 1st Edition, 2002.
   Applied Linear Statistical Models, Michael H. Kutner, Christopher J. Nachtsheim, John Neter and William Li, McGraw-Hill Education, 5th Edition, 2004.

NPTEL/ YouTube/ Faculty Video Link:

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. *Predictive Analytics using R 3.5 (2 hr 53 m)* 2. *Introduction to R (15 hr)* 3. *Statistics for R (9 hr 35 m)*	27 hr 28 min
Video Lectures	1. https://youtu.be/iz-PtN2aVbI 2. https://youtu.be/Cx8Xie5042M 3. https://youtu.be/Cx8Xie5042M 4. https://www.youtube.com/watch?v=fxx_E0ojKr 5. https://www.youtube.com/playlist?list=PLZoTAELRMXVPS-dOaVbAux22vzqdgoGhG	59 min
Web References	1. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.smcedn.edu.in/LMS/docs/PREDICTION%20AND%20REGRESSION.pdf 2. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://aitsrajampet.ac.in/images/pdf/mba/learning-management-system/UNIT4-CORRELATION-AND-REGRESSION-ANALYSIS-min.pdf	10 hrs.

	3. https://www.researchgate.net/publication/388053224_12_Correlation_and_linear_regression 4. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://rajeshreddycse.wordpress.com/wp-content/uploads/2023/09/ml-unit-3-notes-1.pdf 5. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.columbia.edu/~mh2078/MachineLearningORFE/Regression_II.pdf chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://ftp.idu.ac.id/wp-content/uploads/ebook/ip/REGRESI%20LOGISTIK/Practical%20Guide%20to%20Logistic%20Regression%20(%20PDFDrive%20).pdf	

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Course Code: BCSAI0512	Course Name: Advanced Java Programming	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(CYS)/CSE(IOT)/MCT/CSE/CSE-R/IT/M.Tech. (INT).		3	0	0	3

Pre-requisite: Basic knowledge of Core Java (OOP, exception handling, collections), multithreading, and fundamentals of SQL/database operations.

Course Objectives:
To equip students with in-depth knowledge of server-side programming using Java technologies such as Servlets, JSP, JDBC, and frameworks like Spring and Hibernate. The aim of this course is to enable students to build dynamic web applications and enterprise-level solutions with real world project exposure.

Course Outcome: After completion of the course, the student will be able to		Bloom's Level
CO1	Apply JDBC to integrate Java applications with relational databases for dynamic data handling and managing server-side programming using Servlets for handling web requests and responses.	K3
CO2	Analyze the use of JSP scripting elements, expression language, and directives to determine their effectiveness in dynamic web page rendering and maintainability.	K4
CO3	Implement modular, maintainable Java applications using advanced dependency injection techniques within the Spring Ecosystem.	K3
CO4	Design modular, loosely coupled web applications by implementing the MVC architecture using Spring MVC and Spring Boot.	K6
CO5	Deploy JPA to map, store, retrieve, and update data from java objects to relational databases and vice versa with RESTful APIs to enable scalable and maintainable services.	K6

CO-PO Mapping (Scale: 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	2	1	1	1	1	2	3	2	1
CO2	3	2	3	2	2	2	1	1	1	1	2	3	2	1
CO3	3	3	3	2	3	2	1	1	1	1	2	3	2	1
CO4	3	3	2	3	3	1	2	2	2	1	2	2	1	1
CO5	3	3	3	3	3	1	2	2	2	2	3	3	2	1

Course Contents / Syllabus

Unit 1	JDBC & Servlets	9 hours

Jdbc: Introduction, JDBC Driver, DB Connectivity, Connection, Statement, Result Set, Prepared Statement, Transaction Management, Stored Procedures.

Servlet: Servlet Overview, Servlet Life Cycle, Servlet API, Generic Servlet, HTTP Servlet, Redirect requests to other resources using Request Dispatcher, send Redirect, Session Tracking: Using cookies, URL rewriting, and HTTP Session.

Unit 2	JSP	9 hours

JSP: Introduction, Life Cycle of JSP, JSP to Servlet Conversion, JSP Scripting Elements , JSP Implicit Objects, JSP Directives, Expression Language, Exception Handling in JSP, Servlet-JSP-JDBC Integration, Login and registration system using JSP and Servlet.

Unit 3	Spring Framework	9 hours

Spring: Overview of Spring Ecosystem, Spring Units, IOC container, Spring Project Setup using Maven, Types of Dependency Injection (DI): Constructor Injection Setter Injection, Field Injection, Configuration Approaches: Java-based Configuration, Component Scanning, Annotation-based Configuration, Spring JDBC.

Unit 4	Spring MVC & Spring Boot	9 hours

Spring MVC: Overview of Spring MVC architecture, Controllers in Spring MVC, Passing Data Between Controller and View, JSP in Spring MVC, Integration with Spring JDBC.

Spring Boot: Introduction, Creating a Spring Boot project using Spring Initializer, Spring Boot Annotations & Auto Configuration, Spring Data JPA and H2 setup, Serving HTML pages and static content, Handling HTTP methods, Project Lombok		
Unit 5	**JPA**	**9 hours**
JPA: Introduction to ORM & JPA, JPA Annotations, JPA Relationships, RESTful API, CRUD with Spring Data JPA, Repository Interfaces: JpaRepository, CrudRepository, Exception Handling in REST APIs, HTTP Status Code Management, API Versioning Techniques, JSON/XML handling, Postman testing and deployment, Swagger Open API design and documentation tool.		
Total Lecture Hours		**45 hours**

Textbook:

S.No	Book Title	Author
1	Head First Servlets and JSP, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2008.	B. Basham, K. Sierra, and B. Bates
2	Java Server Pages, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2003.	H. Bergsten
3	Spring in Action, 6th ed. Shelter Island, NY, USA: Manning Publications, 2022.	C. Walls
4	Spring Boot in Action, 1st ed. Shelter Island, NY, USA: Manning Publications, 2016.	C. Walls
5	Spring Data: Modern Data Access for Enterprise Java, 1st ed. Sebastopol, CA, USA: O'Reilly Media, 2012.	M. Pollack and O. Gierke,

Reference Books:

S.No	Book Title	Author
1	Core Servlets and JavaServer Pages, Volume 1: Core Technologies, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2003.	M. Hall and L. Brown
2	Core Servlets and JavaServer Pages, Volume 2: Advanced Technologies, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2004.	M. Hall and L. Brown,
3	Pro Spring 6, 1st ed. New York, NY, USA: Apress, 2023.	C. Ho, R. Harrop, and C. Schaefer,
4	Pro JPA 2 in Java EE 8, 3rd ed. New York, NY, USA: Apress, 2018.	M. Keith and M. Schincariol

Certifications:

S.NO	Industry Aligned Certificate
1	**Oracle Academy Certification on JAVA and Database**
2	**Wipro TalentNext Certification on Full Stack**

NPTEL/ YouTube/ Faculty Video Link:

Unit 1	https://www.youtube.com/playlist?list=PLlhM4lkb2sEjVsbbZ_kiixY5CcR84IQUg
Unit 2	https://www.youtube.com/playlist?list=PL-XjHn7CHrmQhMkVC6KCfmsPHdklvUlQr
Unit 3	https://www.youtube.com/playlist?list=PL9ooVrP1hQOEfi91PCFQMawtBJrPpir7y
Unit 4	https://www.youtube.com/playlist?list=PL-XjHn7CHrmQhMkVC6KCfmsPHdklvUlQr
Unit 5	https://www.youtube.com/playlist?list=PLGRDMO4rOGcNSBOJOlrgQqGpIgo6_VZgR

Course Code: BCSDS0513	Course Name: Product Analytics	L	T	P	C
Course Offered in: CSE (Data Science)		3	0	0	3

Pre-requisite: Python, R, Basic Statistics and Excel.

Course Objectives: To understand how users interact with a product. To collect and analyze data related to product usage and performance. To gain insights into user behavior and customer preferences. To support data-driven decision making for product development. To optimize products and improve user experience through analytics.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand product analytics concepts, user behavior, and data-driven decision making.	K2
CO2	Apply statistical methods and A/B testing for product performance analysis.	K3
CO3	Develop predictive models for user segmentation and personalization.	K3
CO4	Analyze causal inference techniques for product optimization and experimentation.	K4
CO5	Evaluate product analytics solutions using modern tools and real-world applications.	K5

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	1	-	-	-	1	1	1	3	2	1
CO2	-	-	-	-	-	-	-	-	-	-	-	3	3	1
CO3	3	3	2	-	-	-	-	-	1	1	1	3	3	2
CO4	3	3	3	-	-	-	-	-	1	1	2	3	2	2
CO5	2	3	2	1	-	-	-		1	1	-	3	3	3

Course Contents / Syllabus

Unit 1	**Qualitative Methodology**	**08 hours**

Introduction to Product Analytics, Role of Data Science in Product Development, Product Lifecycle and Data-Driven Decision Making, User Data Collection and Data Pipelines, Understanding User Behaviour and Customer Journey, Key Product Metrics: DAU, MAU, Retention, Churn, Conversion Rate, Event Tracking and Funnel Analysis, Behavioural Analytics and Actionable Insights, Ethics, Privacy, and Data Governance in Product Analytics. **Hands-on Sessions:** User event tracking, Funnel analysis using sample datasets, Visualization of product metrics.

Unit 2	**Statistical Methods for Product Analytics**	**08 hours**

Descriptive Statistics and Data Exploration, Probability Distributions in Product Data, Correlation and Covariance, Hypothesis Testing, Confidence Intervals and Statistical Significance, A/B Testing and Multivariate Testing, KPI Design and Metric Evaluation, Cohort Analysis and Retention Analysis, Time-Series Basics in Product Analytics. **Hands-on Sessions:** A/B testing implementation, Cohort analysis dashboards.

Unit 3	**Predictive Analytics and Machine Learning**	**09 hours**

Introduction to Predictive Analytics, Data Preprocessing and Feature Engineering, Regression Techniques for Product Analytics, Classification Algorithms, Decision Trees and Random Forests, Clustering Techniques: k-Means and Hierarchical Clustering, Recommendation Systems Basics, User Segmentation and Personalization, Model Evaluation Metrics: Precision, Recall, F1 Score, ROC-AU. **Hands-on Sessions:** Customer churn prediction, User segmentation project.

Unit 4	**Causal Inference and Experimentation**	**10 hours**

Introduction to Causal Inference, Correlation vs Causation, Experimental Design in Product Analytics, Difference-in-Difference Modeling, Regression Discontinuity, Propensity Score Matching, Uplift Modeling, Growth Analytics and Product Optimization, Product Experimentation Frameworks. **Hands-on Sessions:** Causal impact analysis, Experiment design case study.

Unit 5	**Tools, Platforms, and Industry Applications**	**10 hours**

Product Analytics Tools Overview, Data Visualization and Dashboarding, SQL for Product Analytics, Python/R for Product Data Analysis, Analytics Platforms: Google Analytics, Mixpanel, Amplitude. Real-Time Analytics and Big Data Concepts, Product Analytics in E-commerce, OTT, and Social Media. Case Studies: Netflix, Spotify, Instagram, Emerging Trends: AI-driven Analytics and Explainable Analytics. **Hands-on Sessions:** Dashboard creation, SQL-based analytics queries, End-to-end mini project on product analytics.

	Total Lecture Hours	**45 hours**

Textbook:

1. Joanne Rodrigues, Product Analytics: Applied Data Science Techniques for Actionable Consumer Insights, Addison-Wesley Professional, 202
2. Alistair Croll & Benjamin Yoskovitz, Lean Analytics, O'Reilly Media.

Reference Books:

1. Wes McKinney, Python for Data Analysis, O'Reilly Media.
2. Gareth James et al., An Introduction to Statistical Learning, Springer.
3. Galit Shmueli et al., Data Mining for Business Analytics, Wiley.

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. https://shorturl.at/qdAdJ 2. https://shorturl.at/o2vbO	60 Hrs
Video Lectures	1. https://www.youtube.com/@freecodecamp 2. https://www.youtube.com/@statquest	80 Hrs
Web References	1. https://support.google.com/analytics?sjid=4346884306243303523-NC#topic=14090456 2. https://docs.mixpanel.com/docs/what-is-mixpanel?utm_source=chatgpt.com	40 Hrs

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Course Code: BCSE0514	Course Name: Design Patterns	L	T	P	C
Course Offered in: CSE/CS/CSR-R/M.TECH(INT) /IT//CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)		3	0	0	3

Pre-requisite: Object Oriented Analysis and Design. Data structures and algorithms. Programming Language (C++ or Java)

Course Objectives: The course objective is to familiarize the student with techniques for designing reusable combinations of Java classes and organizing their cooperation to produce modular and maintainable Java programs.

Course Outcome: After completion of the course, the student will be able to

S.No	Course Outcome	Bloom's Level
1	Analyze, classify, and apply appropriate design patterns to solve real-world software problems, understanding their structure and selection principles.	K3
2	Identify, compare, and apply creational design patterns (Factory, Abstract Factory, Builder, Prototype, Singleton) to develop flexible and reusable software components.	K4
3	Analyze, design, and implement structural design patterns to improve flexibility, reusability, and maintainability of software systems.	K4
4	Analyze, design, and apply behavioral design patterns (Mediator, Memento, Observer) to manage object interactions and system behavior effectively.	K3
5	Analyze and implement advanced behavioral design patterns to solve common software design problems.	K4

CO-PO Mapping (Scale: 1: Low, 2: Medium, 3: High)

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO	PSO3
CO1	3	3	2	2	2	-	-	1	1	-	2	3	3	1
CO2	3	2	3	1	3	-	-	1	1	-	2	3	3	1
CO3	3	3	3	2	3	-	-	1	1	-	2	3	3	1
CO4	3	3	3	2	2	-	-	1	1	-	2	3	3	2
CO5	3	2	3	2	2	-	-	1	2	-	3	3	3	2

Course Contents / Syllabus

Unit 1	Introduction to Design Patterns	9 hours

Describing Design Patterns, Design Patterns in Smalltalk MVC, The Catalog of Design Patterns, Organizing the Catalogue, Design Patterns for Solving the Real-life Problems, Selection and Use of Design patterns.

Unit 2	Creational Design Pattern	9 hours

Creational Patterns: Abstract Factory, Builder, Factory Pattern, Prototype Pattern, Singleton pattern.

Unit 3	Structural Design Pattern	9 hours

Structural Pattern Part-I, Adapter, Bridge, Composite.
Structural Pattern Part-11, Decorator Pattern, Facade Pattern, Flyweight Pattern, Proxy Pattern.

Unit 4	Behavioral Patterns – I	9 hours

Behavioral Patterns Part: I, Chain of Responsibility Pattern, Command Pattern, Interpreter Pattern, Iterator Pattern. Behavioural Patterns Part: II, Mediator, Memento, Observer Pattern.

Unit 5	Behavioral Patterns – II	8 hours

Behavioral Patterns Part: III, State Patterns, Strategy, Template Patterns, Visitor, Expectation from Design Patterns Relationship between different Design Patterns.

Total Lecture Hours: 45 hours

Textbook:

1. Eric Freeman, Elisabeth Freeman, Kathy Sierra, Bert Bates Head First Design Patterns, 2004, O'Reilly
2. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides Design Patterns: Elements of Reusable Object oriented Software Addison-Wesley, 1995.

Reference Books:

1. Design Patterns Pearson Education by Erich Gamma
2. Patterns in JAVA Volume -I by Mark Grand, Wiley Dream

NPTEL/ YouTube/ Faculty Video Link:

Unit 1	https://www.youtube.com/watch?v=ysoC8S96pZU&list=PLBBog2r6uMCResrR48HziFtNx4-BDOmU0
Unit 2	https://www.youtube.com/playlist?list=PLmCsXDGbJHdhULIoYzIIIeFFJSo1NlMbN
Unit 3	https://www.youtube.com/watch?v=dCWuXDBRFtk&list=PL2NZAYdLkYvhFnqbvLbwf4ZM4rH99Jap6
Unit 4	https://www.youtube.com/watch?v=a4u_caEWTR8&list=PL2NZAYdLkYvg 5SpIDMlRopz-dD_KsIX
Unit 5	https://www.youtube.com/watch?v=a4u_caEWTR8&list=PL2NZAYdLkYvg 5SpIDMlRopz-dD_KsIX

Mode of Evaluation

CIE						ESE	Total
ST1	ST2	ST3	TA1 5	TA2 5	Attendance 5		
35			15			100	150

LAB Course Code: BCSE0552A	LAB Course Name: Computer Networks Lab	L	T	P	C
Course Offered in: CSE/CS/IT /CSE(AI)/CSE(AIML)/ CSE(DS)/ CSE(CYS)/MCT /ECE		0	0	4	2

Pre-requisite: Fundamentals of Computers.

Course Objectives: Understand and configure basic networking concepts using CISCO Packet Tracer and device. Develop practical skills in IP addressing, Subnetting, V LAN, Routing, Switching and wireless networking. Learn to configure and manage network services such as DHCP, NAT, ACLs, and inter VLAN Routing. Perform network testing, Troubleshooting and connectivity verification using CISCO IOS CLI commands and diagnostic tools Prepare students for industry-oriented networking rules and certification such CISCO CCNA by providing hands-on networking experience.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Model computer network through cabling, assigning IP addresses along with subnetting using CISCO packet tracer.	K4
CO2	Analyse Layer 2 and Layer 3 networking features including switching, VLANs, trunking, inter- VLAN routing, STP, EtherChannel, static routing, and basic dynamic routing.	K4
CO3	Demonstrate core network services and security features such as wireless connectivity, DHCP,NAT/PAT, ACLs, transport/application layer protocols, and connectivity verification using network diagnostic tools.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	3	-	-	-	2	2	1	3	2	1
CO2	3	3	3	2	3	1	-	-	2	2	1	3	2	2
CO3	3	3	2	3	3	1	-	1	2	2	1	3	3	2

S. No	List Of Practical's (Indicative & Not Limited To)	CO
1	Explore the CISCO Packet Tracer.	**CO1**
2	Explore Network Devices and Cabling: Identify different network devices (router, switch, PC) and understand cable types (straight-through, crossover, fiber).	**CO1**
3	Basic Switch and End Device Configuration: Configure hostname, passwords, and basic security settings on switches and PCs.	**CO1**
4	Build a Simple Network Using Packet Tracer: Create a small network and verify connectivity using ping and traceroute	**CO1**
5	Configure IPv4 Addressing: Assign IP addresses manually and test communication between devices.	**CO1**

6	Subnetting and Address Calculation: Apply subnetting techniques to divide networks efficiently.	**CO1**
7	Configure IPv6 Addressing: Assign IPv6 addresses and test connectivity.	**CO1**
8	Access Cisco IOS and CLI Navigation: Use CLI commands to configure and manage network devices.	**CO1**
9	MAC address and ARP analysis: understand how MAC address and ARP work in local communication.	**CO1**
10	Basic Switch configuration configure switch settings, ports, and management IP.	**CO1**
11	VLAN Configuration: Create VLANs and assign ports to segment network traffic.	**CO2**
12	Inter-VLAN Routing: Enable communication between VLANs using router or Layer 3 switch.	**CO2**
13	Trunking Configuration (802.1Q): Configure trunk links to allow multiple VLAN traffic.	**CO2**
14	Spanning Tree Protocol (STP): Understand STP operation and prevent switching loops.	**CO2**
15	Static Routing Configuration: Configure static routes and verify packet forwarding.	**CO2**
16	Default and Dynamic Routing (RIP): Implement basic dynamic routing and default routes.	**CO2**
17	Wireless Network Configuration: Configure a wireless router and connect wireless clients.	**CO3**
18	NAT (Network Address Translation): Configure NAT/PAT for internet access.	**CO3**
19	Network Troubleshooting: Identify and resolve connectivity issues using commands like ping, traceroute, and show.	**CO3**
	Total Hours	**60Hrs.**

LAB Course Code: BCSDS0551	LAB Course Name: DevOps and Cloud Computing Lab	L	T	P	C
Course Offered in: CSE (Data Science)		0	0	2	1

Pre-requisite: Basic knowledge of programming, operating systems, networking, Linux commands, Git, and introductory cloud computing concepts.

Course Objectives:

This course provides knowledge of cloud computing concepts, AWS services, and Linux commands for system setup and resource management, along with version control and build automation tools for software development and integration, enabling students to develop and deploy applications using automated tools.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply cloud computing concepts using AWS services and Linux commands to set up systems and manage resources.	K3
CO2	Apply version control and build automation tools for software development and Integration	K3
CO3	Develop and deploy applications using automated tools.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	1	-	3	-	-	1	1	1	2	3	2	1
CO2	2	2	2	1	3	-	1	2	2	2	2	2	3	2
CO3	3	3	3	2	3	1	1	2	2	2	3	3	3	3

List Of Practical's	CO Mapping
1. Execute basic Linux commands and explore the AWS Management Console.	CO1
2. Launch and configure an EC2 instance and connect using SSH.	CO1
3. Create and manage S3 buckets and configure IAM users, roles, and policies.	CO1
4. Write a Python script to upload/download files to an S3 bucket using boto3.	CO1
5. Write a script to launch and stop an EC2 instance using AWS CLI/SDK.	CO1
6. Install Git and perform basic operations (commit, push, pull).	CO2
7. Create a GitHub repository and perform branching, pull requests, and merge operations.	CO2
8. Build a sample project using Maven or Gradle to generate artifacts.	CO2
9. Install Jenkins and integrate it with a GitHub repository.	CO2
10. Develop a Jenkins pipeline to automate build and test processes.	CO3
11. Create Docker images and run containers using Dockerfile.	CO3
12. Deploy a containerized application using Kubernetes (Minikube or cloud).	CO3
13. Create a CI pipeline using GitHub Actions for a sample application.	CO3
14. Write a Docker Compose file for a multi-container application (web + database).	CO3
15. Create a simple CI/CD pipeline integrating GitHub, build tools, Docker, and deployment	CO3
	Total Hours: 30 hrs.

LAB Course Code: BCSDS0516P	LAB Course Name: Predictive Analytics Lab	L	T	P	C
Course Offered in: CSE (DS)		0	0	2	1

Pre-requisite: Students should have basic knowledge of statistics, regression concepts, and hypothesis testing, along with fundamental programming skills in Python or R. They should be familiar with data handling, visualization, and using tools like pandas, numpy, and matplotlib, as well as have basic mathematical understanding and the ability to interpret analytical results.

Course Objectives: This course helps students understand the fundamental concepts of predictive analytics, regression analysis, and time series forecasting techniques, apply Simple Linear Regression, Multiple Linear Regression, and regression-based predictive models for solving real-world problems, and develop skills in pre-processing, feature engineering, evaluation, and interpretation of predictive models using temporal and non-temporal datasets.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply Simple and Multivariate Linear Regression to make predictions.	K3
CO2	Apply regression-based predictive analytics techniques to build and optimize predictive models for real-world problems.	K3
CO3	Apply time series techniques including pre-processing and feature engineering to build, evaluate, and interpret predictive models for real-world data	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	3	3	3	-	-	1	2	-	2	3	3	3
CO2	2	3	3	3	3	-	-	1	3	-	2	3	3	3
CO3	3	3	3	3	3	-	-	1	2	-	2	3	3	3

List Of Practical's (Indicative & Not Limited To)	CO Mapping
1. Write a program in python to apply Simple Linear Regression and Multivariate Linear Regression	CO2
2. Write a program in python to compute covariance and correlation between different variables in a dataset in order to understand the strength and direction of their relationships.	CO2
3. Write a program in python to apply Akaike Information Criterion (AIC) for comparing different regression models and selecting the most appropriate set of features.	CO2
4. Write a program in python to compute the performance and goodness of fit of regression models using statistical measures such as R-square and Adjusted R-square.	CO2
5. Write a program in python to apply predictive regression models on real-world datasets (House prediction price) for forecasting and decision-making purposes.	CO2
6. Write a program in python to show the effect of increasing polynomial degree on model performance.	CO2
7. Write a program in python to compare performance of Linear, Polynomial, Ridge, Lasso, and Elastic Net models on the real word dataset.	CO2
8. Write a program in python to identify the best model based on error metrics and justify your choice.	CO3
9. Write a program in python to compute and visualize correlation matrix to identify relationships between variables.	CO3
10. Write a program in python to show single and double exponential smoothing techniques.	CO3
11. Write a program in python to implement SARIMA model for seasonal data and compare with ARIMA.	CO3
12. Write a program in python to apply a predictive model and evaluate the results using appropriate metrics.	CO3
	Total Hours: 30 hrs.

LAB Course Code: BCSAI0512P	LAB Course Name: Advanced Java Programming Lab	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)/CYS/IOT/MCT		0	0	2	1

Pre-requisite: PROGRAMMING LANGUAGE PREFEREBALLY JAVA

Course Objectives:

Provide hands-on experience to develop scalable, database-driven applications using Spring Boot, JPA, and RESTful APIs with proper integration and deployment practices.

Course Outcome: After completion of the course, the student will be able to

CO1. Students will be able to develop dynamic web applications using JDBC, Servlets, and JSP with proper database connectivity and session management.

CO2. Students will apply Spring Framework concepts, including IOC and Dependency Injection, to build structured and scalable applications.

CO3. Students will design and implement RESTful web services using Spring Boot and JPA, and deploy complete full-stack applications effectively.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**3**	3	2	2	1	1	-	3	1	1	2	3	2	1
CO2	3	3	3	2	2	1	-	3	1	1	2	3	2	1
CO3	3	3	3	2	2	1	-	3	1	1	2	3	3	2

List Of Practical's (Indicative & Not Limited To)

1. Write Java program to create a demo table emp having fields is, name, city and insert two rows by using JDBC.
2. Write a Java program using Statement to create a table Students with fields id, name, age, and grade.
3. Write a Java program to demonstrate a money transfer transaction between two bank accounts. Ensure that both debit and credit operations are either committed together or rolled back in case of an error.
4. Write a Java program using PreparedStatement to delete student records where grade is below a given threshold.
5. Write a Java program to call a stored procedure that calculates the total salary of employees in a given department.
6. Write a servlet program to select the details of an employee (emp id, empname, empadd,empphone) and display on browser in appropriate format.
7. Write a GenericServlet to handle employee registration. The servlet should: Accept employee details (name, email, designation, salary) via a form (POST request).Store the details in a database (use JDBC). Display a success message after successful insertion
8. Implement a servlet program that redirect a request to Google.com.
9. Develop a Java Servlet program to demonstrate session tracking using the following techniques:

   (i) Cookies

(ii) URL Rewriting

(iii) HttpSession

The program should maintain user information (such as visit count or username) and display it across multiple requests using all three session tracking methods.

10. Implement all JSP scripting element (scriptlet, expression and declaration).
11. Implement all JSP directive element (page, include and taglib).
12. Implement a JSP program for arithmetic exception error handling by using html page, jsp page, and errorPage, isErrorPage directives.
13. A company's recruitment portal requires candidates to register before applying for jobs. Design a JSP page for registration and a Servlet to store user details in a MySQL database using JDBC. How would you implement session management in a JSP-based web application to keep users logged in across multiple pages?
14. Implement a JSP which insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user's name and password from the database. Design and implement a simple shopping cart example with session tracking API.
15. Demonstrate the use of ApplicationContext as IOC container to load beans. Concept: Spring IOC container usage.
16. Define multiple Spring Beans and manage them using XML-based configuration. Concept: Spring Units and IOC in action.
17. Implement Setter-based DI to inject a service bean into another. Concept: Using setters for bean wiring.
18. Demonstrate Field Injection using @Autowired annotation. Concept: Simplified DI using annotations.
19. Program: Configure beans using Java-based configuration with @Configuration and @Bean. Concept: Pure Java Spring configuration.
20. Program: Use Component Scanning and @Component, @Service, @Repository. Concept: Auto-detection of components.
21. Program: Set up a Maven project structure for a spring app and manage dependencies using pom.xml. Concept: Maven-based project setup
22. You are designing a web application for an online bookstore. Set up a basic Spring MVC structure with Dispatcher Servlet, controller, and view." Program Goal: Create a minimal Spring MVC app showing folder structure and web.xml setup.
23. Create a basic Spring Boot application using Spring Initializer and write a program to display "Hello World" on the browser using appropriate Spring Boot annotations.
24. Develop a Spring Boot application that uses annotations like @SpringBootApplication and @RestController to create a simple REST API that returns a welcome message.
25. Build a feedback page for a college portal. Create a Spring MVC controller that returns a welcome message on accessing /feedback." Program Goal: Implement a basic controller class using @Controller and map a simple GET request.
26. Develop a Spring Boot web application for a library website that displays a homepage containing contact information and a static banner. The application should serve HTML pages from the /templates directory and load static resources such as images and CSS from the /static directory to demonstrate proper handling of dynamic views and static content.

Total Hours: 30Hrs
Total Hours: 30Hrs
Total Hours: 30Hrs

27. Develop a Java program for a Book management system where you need to reduce boilerplate code (such as getters, setters, and constructors) in the Book class. Use Project Lombok annotations like @Data, @NoArgsConstructor, and @AllArgsConstructor to simplify the class definition. The program should create Book objects and display their details to demonstrate the use of these annotations.
28. You are designing a simple library system. Begin by creating an entity Book and use JPA to persist data to a relational database.
29. In a school database system, annotate an @Entity class Student with appropriate JPA annotations like @Id, @GeneratedValue, @Column.
30. In a hospital management system, create a one-to-one relationship between Patient and Medical Record using JPA
31. Implement CRUD operations on MYSQL database using spring data rest with POSTMAN client.
32. Develop a system for online courses where one Instructor can have many Courses. Model this using JPA annotations.
33. Develop a simple Java application using JPA to create an entity class (e.g., Student) and map it to a database table using basic JPA annotations like @Entity, @Id, and @Column.
34. Write a program using JPA to perform basic CRUD operations (insert and display) on an entity using annotations and an ORM approach.
35. Create a JPA-based application that demonstrates the use of annotations such as @Table, @GeneratedValue, and @Column for mapping an entity to a relational database.
36. Build a college management system where Students can enroll in multiple Courses. Implement many-to-many using JPA.
37. Develop an employee management system that performs Create, Read, Update, and Delete operations using JpaRepository.
38. In a book store app, compare the use of CrudRepository vs. JpaRepository for managing Book entities
39. Develop a RESTful web service that produces and consumes data in both JSON and XML formats, and test the API using Postman.
40. Create a Simple application with a REST API for CRUD operations, demonstrate JSON/XML data handling, test it using Postman, and deploy the application on a local server.

Total Hours: 30Hrs

Course Code: BSCML0555B	Course Name: Deep Learning for Data Analytics	L	T	P	C
Course Offered in: CSE (Data Science)		0	0	6	3

Prerequisite: Basic knowledge of discrete mathematics, programming fundamentals, computer networks, and modular arithmetic.

Course Objectives: Knowledge of the concepts and working principles of modern learning models and methods used for evaluating their effectiveness. Application of models for image analysis, classification, and visual data interpretation tasks. Exploration of advanced techniques for identifying, recognizing, and analyzing objects and patterns in digital images. Learning methods for processing sequential data and applying them to language-related and predictive applications. Evaluation of models for feature learning, data representation, and improving the accuracy and efficiency of predictive systems.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Analyze ANN model and understand the ways of accuracy measurement.	K4
CO2	Develop a convolutional neural network for multi-class classification in images.	K6
CO3	Apply Deep Learning algorithm to detect and recognize an object.	K3
CO4	Apply RNNs to Time Series Forecasting, NLP, Text and Image Classification.	K3
CO5	Apply Lower-dimensional representation over higher-dimensional data for dimensionality reduction and capture the important features of an object.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	3	2	2	1	-	1	-	2	2	3	3
CO2	3	3	3	3	2	2	1	-	1	1	2	2	3	3
CO3	3	3	3	3	3	2	2	-	2	1	2	3	3	3
CO4	3	3	3	3	3	2	2	1	2	1	2	3	3	3
CO5	3	3	3	3	3	2	2	1	2	1	2	2	3	3

Course Contents / Syllabus

Unit 1	Introduction	16 hours

Introduction:

Model Improvement and Performance: Curse of Dimensionality, Bias and Variance Trade off, Overfitting and underfitting, Regression - MAE, MSE, RMSE, R Squared, Adjusted R Squared, p-Value, Classification - Precision, Recall, F1, Other topics, K-Fold Cross validation, RoC curve, Hyper-Parameter Tuning Introduction – Grid search, random search, Introduction to Deep Learning.

Artificial Neural Network: Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: Single layer and Multilayer feed forward networks, recurrent networks. Various learning techniques; Perception and Convergence rule, Hebb Learning. Perceptron's, Multilayer perceptron, Gradient descent and the Delta rule, Multilayer networks, Derivation of Backpropagation Algorithm.

Unit 2	CONVOLUTION NEURAL NETWORK	10 hours

What is computer vision? Why Convolutions (CNN)? Introduction to CNN, Train a simple convolutional neural net, Explore the design space for convolutional nets, Pooling layer motivation in CNN, Design a convolutional layered application, Understanding and visualizing a CNN, Transfer learning and fine-tuning CNN, Image classification, Text classification, Image classification and hyper-parameter tuning, Emerging NN architectures.

Unit 3	DETECTION & RECOGNITION	14 hours

Padding & Edge Detection, Strided Convolutions, Networks in Networks and 1x1Convolutions, Inception Network Motivation, Object Detection, YOLO Algorithm. Different types of RNNs, Introduction to Language model and sequence generation		
Unit 4	**RECURRENT NEURAL NETWORKS**	**16 hours**
Why use sequence models? Recurrent Neural Network Model, Notation, Back-propagation through time (BTT), Different types of RNNs, Language model and sequence generation, Sampling novel sequences, Vanishing gradients with RNNs, Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM), Bidirectional RNN, Deep RNNs		
Unit 5	**AUTOENCODERS AND ADVANCED NEURAL NETWORKS IN DEEP LEARNING**	**16 hours**
Auto-encoders and unsupervised learning, Stacked auto-encoders and semi-supervised learning, Regularization - Dropout and Batch normalization.,Generative Adversarial Networks (GANs),Residual Networks (Res Net)		
	Total Lecture Hours	**72 hours**

Textbook:

1. Zurada and Jacek M, "Introduction to Artificial Neural Systems", West Publishing Company, 1992,

   ISBN: 9780534954604

2. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.

3. Simon Haykin, "Neural Networks and Learning Machines" Third Edition

4. Deep Learning", I Goodfellow, Y Bengio and A Courville, 1st Edition 2016

5. Introduction to Machine Learning with Python ", by Andreas C. Müller, Sarah Guido

6. R2. Deep Learning with Python by François Chollet 1st Editions**, 7th Edition Pearson**

Reference Books:

1. Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola "Dive into Deep Learning", Release0.17.4
2. Artificial Intelligence: A Modern Approach. Prentice Hall Series in Arti Russell, S. and Norvig, N. ArtiIntelligence. 2003.

NPTEL/ YouTube/ Faculty Video Link:

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01278210 5116811264219_shared/overview 2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01274814 254931148859_shared/overview	39hrs

Video Lectures	371) Lec-1 Introduction to Artificial Neural Networks - YouTube (3) Deep Learning(CS7015): Lec 8.1 Bias and Variance - YouTube (3) Mod-10 Lec-39 Assessing Learnt classifiers; Cross Validation; - YouTube (3) Lec-1 Introduction to Artificial Neural Networks - YouTube (3) Lec-2 Artificial Neuron Model and Linear Regression - YouTube (3) Evaluation and Cross-Validation - YouTube (3) Lecture 1 | Introduction to Convolutional Neural Networks for Visual Recognition - YouTube (3) Lecture 2 | Image Classification - YouTube	6hrs
Web References	https://introtodeeplearning.com/	
Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.		

List Of Practical's	CO Mapping
1. Write a program Print Dimensions of dataset.	**CO1**
2. Write a program to Calculate of Accuracy Values.	CO1
3. Write a program to Build an Artificial Neural Network by implementing the Backpropagation a lgorithm and test the same using appropriate data sets.	CO1
4. Write a program to Compose Matrix Shape and Tensor Shape.	CO1
5. Write a program to showing Accessing and Manipulation of tensors.	CO1
6. Write a program to showing Accessing and Manipulation of tensors.	CO2
7. Write a program to understand the mechanism of practically training a binary classifier.	CO2
8. Implement with a program showing Access and manipulation of tensors.	CO1
9. Write a program to show Regression Data Sampling.	CO2
10. Write a program to Combat Overfitting.	CO1
11. Write a program Print Dimensions of dataset.	CO1
12. Write a program to showing Accessing and Manipulation of tensors.	CO1
13. Write a program to show Regression Data Sampling.	CO1
14. Write a program to Combat Overfitting.	CO1
15. Write a program to Build an Artificial Neural Network by implementing the Backpropa gation algorithm and test the same using appropriate data sets.	CO1
16. Write a program to build a simple autoencoder based on a fully connected layer in Kera's.	CO1
17. Implement Long Short-Term Memory Networks using sample data.	CO1
18. Write a program showing Automatic Image Captioning with Keras --- Facial Recognition.	CO1
19. Write a program to implement Gradient Descent Algorithm for optimization.	CO1

20. Write a program to demonstrate Activation Functions (ReLU, Sigmoid, Tanh).	CO3
21. Write a program to implement a Multilayer Perceptron (MLP) using Keras.	CO3
22. Write a program to perform Image Classification using CNN (Convolutional Neural Network).	CO3
23. Write a program to apply Dropout Regularization in neural networks.	CO3
24. Write a program to implement Batch Normalization.	CO3
25. Write a program to perform Sentiment Analysis using Deep Learning (LSTM).	CO3
26. Write a program to implement a Recurrent Neural Network (RNN) for sequence prediction.	CO3
27. Write a program to perform Time Series Forecasting using LSTM.	CO3
28. Implementation of Perceptron Algorithm for Binary Classification	CO3
29. Design a Multilayer Perceptron using TensorFlow/Keras	CO1
30. Design CNN for Cat vs Dog Classification	CO1
31. Write a program forFeature Extraction using Stacked Autoencoders	CO1

Course Code: BCSCC0501Y	Course Name: Design Thinking –II	L	T	P	C
Course Offered in: CSE(Data Science)		2	0	0	2

Pre-requisite: Student must complete Design Thinking-I course.

Course Objectives: The objective of this course is to upgrade Design Thinking skills by learning & applying advanced and contextual Design Thinking Tools. It aims to solve a Real-Life Problem by applying Design Thinking to create an impact for all the stakeholders

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Learn sophisticated design tools to sharpen their problem-solving skills	K2
CO2	Construct innovate ideas using design thinking tools and converge to feasible ideas for breakthrough solution	K6
CO3	Implement storytelling for persuasive articulation	K3
CO4	Understanding the nature of leadership empowerment	K2
CO5	Understand the role of a human being in ensuring harmony in society and nature	K2

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2	2	3	-	-	-	-	-	-	1	3	-
CO2	2	2	3	2	2	-	-	-	2	1	1	2	2	1
CO3	-	-	-	-	-	-	-	-	1	3	-	-	-	2
CO4	-	-	-	-	-	-	-	1	3	2	2	-	-	3
CO5	-	-	-	-	-	3	3	2	1	1	-	-	-	3

Course Contents/ Syllabus

Unit 1	Introduction	8 hours

Design thinking & Innovation, Design Thinking Mindset and Principles, recap of 5-Step Process of Design Thinking, Design Approaches, additional in-depth examples of each design approaches. Simon Sinek's – Start with Why, The Golden Circle, Asking the "Why" behind each example (an in-class activity of asking 5-WHYS) , The Higher Purpose, in-class activity for LDO & sharing insights Visualization and its importance in design thinking , reflections on wheel of life (in-class activity for visualization & Wheel of Life), Linking it with Balancing Priorities (in class activity), DBS Singapore and Bank of Americas' Keep the Change Campaign. Litter of Light & Arvind Eye Care Examples, understanding practical application of design thinking tools and concepts, case study on McDonald's Milkshake / Amazon India's Rural Ecommerce & Gillette

Working on 1-hour Design problem, Applying RCA and Brainstorm on innovative solutions. Main project allocation and expectations from the project.

Unit 2	Refinement and Prototyping	10 hours

Refine and narrow down to the best idea, 10-100-1000gm, QBL, Design Tools for Convergence – SWOT Analysis for 1000gm discussion. In-class activity for 10-100-1000gm & QBL.

Prototyping (Convergence): Prototyping mindset, tools for prototyping – Sketching, paper models, pseudo-codes, physical mockups, Interaction flows, storyboards, acting/role-playing etc, importance of garnering user feedback for revisiting Brainstormed ideas. Napkin Pitch, Usability, Minimum Viable Prototype, Connecting Prototype with 3 Laws, A/B Testing, Learning Launch. Decision Making Tools and Approaches – Vroom Yetton Matrix, Shift-Left, Up, Right, Value Proposition, Case study: Careerbuddy, You-Me-Health Story & IBM Learning Launch.

In-class activities on prototyping- paper-pen / physical prototype/ digital prototype of project's 1000gm idea.

Unit 3	Storytelling, Testing and Assessment	8 hours
Storytelling: Elements of storytelling, Mapping personas with storytelling, Art of influencing, Elevator Pitch, Successful Campaigns of well-known examples, in-class activity on storytelling. Testing of design with people, conducting usability test, testing as hypothesis, testing as empathy, observation and shadowing methods, Guerrilla Interviews, validation workshops, user feedback, record results, enhance, retest, and refine design, Software validation tools, design parameters, alpha &beta testing, Taguchi, defect classification, random sampling. Final Project Presentation and assessing the impact of using design thinking.		
Unit 4	**Innovation, quality and Leadership**	**8 hours**
Innovation: Need & Importance, Principles of innovations, Asking the Right Questions for innovation, Rationale for innovation, Quality: Principles & Philosophies, Customer perception on quality, Kaizen, 6 Sigma. FinTech case study of Design Thinking application – CANVAS Leadership, types, qualities and traits of leaders and leadership styles, Leaders vs Manager, Personas of Leaders & Managers, Connecting Leaders-Managers with 13 Musical Notes, Trait theory, LSM (Leadership Situational Model), Team Building Models: Tuckman's and Belbin's. Importance of Spatial elements for innovation.		
Unit 5	**Understanding Human Desirability**	**8 hours**
Comprehensive human goal: the five dimensions of human endeavour (Manaviya - Vyavstha) are: Education-Right living (Sikhsa-Sanskar), Health – Self-regulation (Swasthya - Sanyam), Justice – Preservation (Nyaya-Suraksha), Production – Work (Utpadan – Karya), Exchange – Storage (Vinimya – Kosh), Darshan- Gyan-Charitra (Shifting the Thinking). Interconnectedness and mutual fulfilment among the four orders of nature recyclability and self-regulation in nature, Thinking expansion for harmony: Self-exploration (Johari's window), group behaviour, interpersonal behaviour and skills, Myers-Briggs personality types (MBTI), FIRO-B test to repair relationships.		
	Total Lecture Hours	**42 hours**

Textbook:

S. No	Book Title	Author
1.	UnMukt : Science & Art of Design Thinking, 2020, Polaris	Arun Jain
2.	Basics Design 08: Design Thinking, 2010, AVA Publishing SA	Gavin Ambrose and Paul Harris
3.	A Foundation Course in Human Values and Professional Ethics, First Edition, 2009, Excel Books: New Delhi	R R Gaur, R Sangal, G P Bagaria

Reference Books:

S.No	Book Title	Author
1.	Solving Problems with Design Thinking – Ten Stories of What Works, 2013, Columbia Business School Publishing.	Jeanne Liedta, Andrew King and Kevin
2.	Universal Human Values and Professional Ethics, 2022, Katson Books.	Dr Ritu Soryan
3.	101 Design Methods: A Structured Approach for Driving Innovation in Your Organization, 2013, John Wiley and Sons Inc, New Jersey.	Vijay Kumar
4.	Design of Business: Why Design Thinking is the Next Competitive Advantage, 2009, Harvard Business Press, Boston MA.	Roger L. Martin
5.	Change by Design, 2009, Harper Collins.	Tim Brown

NPTEL/ You tube/ Faculty Video Link:

Unit 1	https://www.youtube.com/watch?v=6_mHCOAAEI8 https://nptel.ac.in/courses/110106124 https://designthinking.ideo.com/ https://blog.experiencepoint.com/how-mcdonalds-evolved-with-design-thinking
Unit 2	https://www.coursera.org/lecture/uva-darden-design-thinking-innovation/the-ibm-story-iq0kE https://www.coursera.org/lecture/uva-darden-design-thinking-innovation/the-meyouhealth-story-part-i-what-is-W6tTs https://onlinecourses.nptel.ac.in/noc19_mg60/preview

Unit 3	https://www.youtube.com/watch?v=HTSCbxSxsg&list=PL1xHD4vteKYVpaIiy295pg6_SY5qznc77&index=5 https://www.youtube.com/watch?v=NnlS2BzXvyM https://www.youtube.com/watch?v=7enWesSofhg
Unit 4	https://youtu.be/rthuFS5LSOo https://www.youtube.com/watch?v=kho6oANGu_A
Unit 5	https://www.youtube.com/watch?v=9vMpHk44XXo&list=PL1xHD4vteKYVpaIiy295pg6_SY5qznc77&index=6 Reinforcement Learning Tutorial | Reinforcement Learning Example Using Python | Edureka - YouTube Association Rule Mining – Solved Numerical Question on Apriori Algorithm(Hindi) - YouTube Q Learning Explained | Reinforcement Learning Using Python | Q Learning in AI | Edureka - YouTube

Mode of Evaluation

CIE								Total
ST1	**ST2**	**ST3**	**TA1 5**	**TA2 5**	**TA3 5**	**Attendance 5**		
60			**40**					**100**

Course Code: BCSCC0552	Course Name: Research Methodology, Ethics and Writing.	L	T	P	C
Course Offered in: All Engineering Programs		0	0	2	1

Pre-requisite: Basic communication skills, engineering fundamentals

Course Objectives:
This course equips students with fundamental research skills, including problem identification, literature review, data analysis, technical documentation, and ethical publication practices, while enhancing technical writing and scientific communication for academic and professional applications.

Course Outcome: After completion of the course, the student will be able to

Course Outcome	
CO1	Analyze research problems, literature, and research methodologies to identify knowledge gaps and formulate research objectives.
CO2	Apply appropriate data collection, analysis, documentation, citation, and ethical research practices in engineering and scientific investigations.
CO3	Develop technical reports, research manuscripts, and professional presentations using scientific writing standards, research integrity principles, and effective communication practices.

CO-PO Mapping (Scale: 1: Low, 2: Medium, 3: High)

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	–	2	1	–	–	2	–	1	–	3	2	1
CO2	2	3	1	3	2	–	–	2	1	2	–	2	3	3
CO3	2	3	2	3	2	–	–	3	2	3	1	1	2	3

Course Contents / Syllabus

Unit 1	Fundamentals of Research Methodology and Literature Review	7 hours

Definition of research, objectives of research, motivation and significance of research, types of research including fundamental, applied, qualitative, quantitative, and experimental research, stages of research process, research problem identification, formulation of research questions, identification of technical gaps, literature review methodologies, use of scientific databases such as Google Scholar, Scopus, Web of Science, IEEE Xplore, and ScienceDirect, keyword selection strategies, source reliability and credibility evaluation, annotated bibliography preparation, critical analysis of published research papers, and review writing techniques.

Unit 2	Data Collection, Analysis, and Technical Documentation	7 hours

Primary and secondary data sources, data gathering methods, survey techniques, questionnaire design, experimental observations, data organization, structured sampling methods including random sampling, stratified sampling, and systematic sampling, basic statistical concepts including mean, median, standard deviation, graphical data representation, data analysis techniques, spreadsheet-based analysis, technical report preparation, scientific report structure including abstract, introduction, methodology, results, discussion, and conclusion, mini-project execution, research documentation, and technical documentation practices.

Unit 3	Citation Management, Publication Ethics, and Research Integrity	7 hours

Ethical research practices, scholarly publishing standards, academic reference management, citation systems including IEEE, APA, and MLA, reference management software such as Mendeley and Zotero, plagiarism awareness, similarity detection tools, responsible citation practices, authorship ethics, predatory journals, peer review systems, journal indexing, impact factors, research integrity principles, publication misconduct, ethical compliance, professional publishing standards, citation formatting exercises, plagiarism analysis, journal quality evaluation, and ethical case studies.

Unit 4	Technical Writing and Research Documentation	9 hours

Scientific and technical writing skills, engineering research documentation, professional documentation practices, technical report writing, project report preparation, review paper writing, conference paper drafting, research manuscript development, scientific writing styles, report organization, abstract preparation, introduction framing, methodology documentation, results presentation, discussion development, conclusion writing, proofreading, technical editing, document formatting, visual representation of

technical data using tables, charts, and figures, documentation standards, formal technical presentations, dissemination of research findings, manuscript section drafting, technical document editing, and professional written communication.

Total Lecture Hours	**30 hours**

Textbook:

1. C.R. Kothari & Gaurav Garg – Research Methodology: Methods and Techniques
2. Ranjit Kumar – Research Methodology: A Step-by-Step Guide for Beginners
3. Michael Alley – The Craft of Scientific Writing
4. Day & Gastel – How to Write and Publish a Scientific Paper
5. George Thomas – Research Methodology and Scientific Writing

Reference Materials

- IEEE Author Guidelines
- Elsevier Researcher Academy Resources
- Springer Author Academy
- WIPO Resources for Research Documentation
- University digital library resources

NPTEL/ YouTube/ Faculty Video Link:

Unit 1	https://www.youtube.com/watch?v=rz30rRfManE&list=PLdj5pVg1kHiOypKNUmO0NKOfvoIThAv4N
Unit 2	https://www.youtube.com/watch?v=HYj4Ght1_qs
Unit 3	https://www.youtube.com/watch?v=N9tJCs-czOU&list=PLnIj1GYixjfrOhcWQq_ANzLFDniGYZ_-Q
Unit 4	https://www.youtube.com/watch?v=NM53k7x_jjk

Mode of Evaluation

CIE						ESE	Total
ST1	**ST2**	**ST3**	**TA1**	**TA2**	**Attendance**		

SYLLABUS- SEMESTER VI

Course Code: BCSDS0601	Course Name: BIG DATA ANALYTICS	L	T	P	C
Course Offered in: CSE (Data Science)		3	1	0	4

Prerequisite: Basic knowledge of database systems and core Java or Python programming. Familiarity with operating systems and command-line interfaces is recommended.

Course Objectives: The course helps students understand Big Data concepts, Hadoop ecosystem components, and tools like Hive, Pig, HBase, and Spark. It develops practical skills to design, manage, and analyze large-scale data processing pipelines.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Recall fundamental concepts of Big Data, its characteristics, and the evolution of data generation.	K1
CO2	Understand Hadoop architecture, HDFS, YARN, and MapReduce programming models.	K2
CO3	Apply Hadoop ecosystem tools like Hive, Pig, and HBase for data storage, querying, and processing.	K3
CO4	Analyze data ingestion workflows and scheduling using Sqoop, Flume, and Oozie within the Hadoop ecosystem.	K4
CO5	Evaluate and compare Big Data processing engines such as MapReduce and Apache Spark for different analytics scenarios.	K5

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	-	-	1	-	-	1	2	1	2
CO2	3	3	1	2	2	-	-	-	-	-	1	2	2	2
CO3	2	2	3	2	3	-	-	-	-	-	1	3	3	2
CO4	2	3	2	3	3	1	1	1	1	2	2	3	3	2
CO5	2	3	2	3	3	-	-	-	-	-	2	3	3	2

Course Contents / Syllabus

Unit 1	INTRODUCTION TO BIG DATA & HADOOP	9 hours

Introduction: Types of Digital Data, History of Big Data Innovation, Introduction and platform, Drivers for Big Data, Big Data architecture and characteristics, Core 5 Vs of Big Data, Big Data technology components, Big Data importance and applications, Big Data features – security, compliance, auditing and protection, Big Data privacy and ethics, Big Data Analytics, Challenges of conventional systems, nature of data, analytic processes and tools, analysis vs reporting, modern data analytic tools.

Hadoop: History of Hadoop, HDFS Design, concepts, benefits and challenges, file sizes, Block storage mechanism in HDFS, Data replication, HDFS store, read, and write files, command-line interface, Hadoop file system interfaces, Data flow, Data ingest with Flume and Scoop, Hadoop archives, Hadoop I/O: compression, serialization, Avro and file-based data structures, analyzing data with Hadoop, scaling out, Hadoop streaming, Hadoop pipes.

Unit 2	HADOOP ARCHITECTURE & MAP REDUCE	9 hours

Hadoop Eco System and YARN: Hadoop ecosystem components, schedulers, fair and capacity, Hadoop 3.- New Features - Name Node high availability, HDFS federation, MRv2, Name node and Data node functionality, replication and fault tolerance in HDFS, data locality and rack awareness, Hadoop configuration and daemons, YARNv2 architecture and resource management.

Map Reduce: Map Reduce framework basics & functionality, developing a Map Reduce application, Anatomy of a Map Reduce job run, failures, job scheduling, shuffle and sort, task execution, Map Reduce types, input formats, output formats, Map Reduce Features.

Unit 3	**Hadoop Ecosystem and Big Data Processing Frameworks**	**9 hours**

PIG: Introduction to PIG, Architecture, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin- Input and output, Relational operators, User defined functions. Working with scripts, Data Processing operators.

Hive: Apache Hive architecture and installation, Hive shell, Hive services, Hive megastore, comparison with traditional databases, HiveQL, tables, querying data and user defined functions, sorting and aggregating, Map Reduce scripts, joins & subqueries.

HBase: HBase vs RDBMS, HBase architecture, advanced indexing, region servers, tables and schema design, CRUD operations, and integration of HBase with HDFS.

Apache Oozie: Workflow scheduling, coordinator jobs, bundle jobs, real-time log collection and ETL design using Flume and Sqoop, integrating pipelines with Hive and HDFS.

Apache Zookeeper: Zookeeper for coordination, monitoring a cluster, Configure and build applications with Zookeeper.

Apache Ambari: Apache Ambari introduction, Installation and architecture of Ambari, Various advantages of Ambari, Configure Ambari to specific Hadoop cluster needs, Work with various components to manage and monitor, Ambari Install Wizard, Work with Ambari web app.

Unit 4	**DATA INGESTION AND INTEGRATION**	**9 hours**

Apache Sqoop: Sqoop Imports, Sqoop - File Formats, Jobs & Incremental Imports, Hive – Exports.

Apache Flume: Introduction to Flume, Replication, Consolidation & Multiplexing, configuration and use cases.

Kafka: Kafka Architecture, Partitions and Offsets, Kafka Consumer/Producers, Kafka Message, Kafka Serialization & Deserialization, Use Cases and Usage.

Unit 5	**REAL TIME PROCESSING**	**9 hours**

Scala: Introduction, classes and objects, basic types and operators, built-in control structures, functions and closures, and inheritance.

Spark: Introduction to Apache Spark as an advanced Big Data processing engine, Spark ecosystem overview, Spark Core, Spark SQL, Spark Streaming, comparison with MapReduce, Spark architecture, driver, executors, DAG scheduler, understanding RDDs and Data Frames, Spark SQL operations and integration with Hive, basics of stream processing with Spark Streaming, Machine learning using Spark MLlib, real-world Big Data pipeline design Integrating Spark, Hive, HBase, and Kafka for real-time and batch analytics.

	Total Lecture Hours	**45 hours**

Textbook:

1. Big Data Black Book (Covers Hadoop 2, MapReduce, Hive, YARN, Pig, R and Data Visualization)
2. Big Data Analytics by Seema Acharya & Subhashini Chellappan (Wiley India)

Reference Books:

1. Big Data Fundamentals: Concepts, Drivers & Techniques by Thomas Erl, et al.
2. Hadoop in Action by Arvind R. & Saurabh Kumar (Dreamtech Press)

NPTEL/ YouTube/ Faculty Video Link:

Self-Study Content Links:		Duration (in Hrs.)
MOOCs	1. Acquiring Data on Nasscom 2. Big Data Analytics Projects with Apache Spark on Infosys SpringBoard	15 hours
Video Lectures	Unit-1: https://www.youtube.com/watch?v=1vbXmCrkT3Y Unit-2: https://youtu.be/ULtOZqlZnCw?si=N8wzzEK29B4ooXzM	50 hours

	https://youtu.be/OP8BsGnqi9c?si=zhR8OvSVAklEDM Ft Unit-3: https://youtu.be/FNt4E2eu1zQ?si=mJf0eGZ8F7yk0Z21 https://youtu.be/IiKhXC6NFDg?si=ovbG8O_YEItilcSs https://youtu.be/HcDvRnYAaHU?si=e0-MxAIIRLGXrsYU Unit-4: https://youtu.be/57TiejstzqI?si=MF9vZhPAP6Sw8CDE https://youtu.be/JFPN-GwON9U?si=WoHDnpqxsj0yKSwf https://youtu.be/X6J33HpIhCc?si=kCgyD_vN64-6G39g Unit-5:https://www.youtube.com/watch?v=9QxZhapbo-o https://www.youtube.com/live/PzQ3DBrHW-U?_x	
Web References	1. https://www.simplilearn.com/resources/big-data-and-analytics/tutorials?utm_source=chatgpt.com 2. https://www.simplilearn.com/resources/big-data-and-analytics/tutorials?utm_source=chatgpt.com	10 hours
Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.		

Course Code: BCSE0603	Course Name: Software Engineering and Design	L	T	P	C
Course Offered in: CSE /CSE-R/ CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		3	1	0	4

Pre-requisite: fundamental knowledge of programming concepts, data structures, object-oriented concepts

Course Objectives: To help students understand all phases of the Software Development Life Cycle (SDLC) both theoretically and practically, enabling them to systematically apply principles of analysis, design, development, testing, and maintenance to build cost- effective software solutions and become competent software engineering professionals.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand software engineering principles, software process models, and Agile methodologies for effective software project development and management.	K2
CO2	Demonstrate SRS documentation and software quality assurance practices to ensure software design and development meet applicable standards.	K3
CO3	Compare and contrast various design concepts to develop effective software design solutions.	K4
CO4	Analyze software testing techniques along with test management practices to ensure and improve software quality.	K4
CO5	Analyze software maintenance, project management, estimation techniques, software metrics, risk management, and configuration management concepts in software development.	K4

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2								1	2		2
CO2	2	2	2		2		2			1	1	3	2	
CO3	2	2	3		2							2	1	3
CO4		2	2	2	3						1	2	2	
CO5			1		2					2	3	1	3	2

Course Contents / Syllabus

Unit 1	Introduction to Software Engineering and Development models	10 hours

Evolving role of software, Software Characteristics, Software crisis, silver bullet, Software myths, Software Engineering Phases, Team Software Process (TSP), Emergence of software engineering, Software process, project and product, Software Process Models: Waterfall Model, Prototype Model, Spiral Model, Iterative Model, Incremental Model, Agile Methodology: Scrum Sprint, Scrum Team, Scrum Master, Product Owner, Kanban framework.

Unit 2	Software Requirement and Quality Assurance	8 hours

Software Requirement Specifications (SRS): Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modelling, Use Case Diagram, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS. Software Quality Assurance (SQA): Quality concepts, SQA activities, Formal approaches to SQA; Statistical software quality assurance; CMM, The ISO standard.

Unit 3	Software Design	10 hours

Design principles, the design process; Design concepts: refinement, modularity: Cohesion, Coupling, Effective modular design: Functional independence, Design Heuristics for effective modularity, Software architecture: Function Oriented Design, Object Oriented Design: OOPs concepts-Abstraction, object, classification, inheritance, encapsulation, UML Diagrams-Class Diagram,
Interaction diagram, Activity Diagram, control hierarchy: Top-Down and Bottom-Up Design, structural partitioning, software procedure.

Unit 4	Software Testing	9 hours

Software Testing: Testing Objectives, 7 Principals of Testing, Levels of Testing: Unit Testing, System Testing, Integration Testing, User Acceptance Testing, Regression Testing, testing for Functionality and Testing for Performance, Top Down and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Structural (White Box Testing Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha, and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards, Test Management, Test Planning and Estimation, Test Monitoring and Control, Configuration Management, Risks and Testing, Defect Management,
Tool Support for Testing, Effective Use of Tools.

Unit 5	**Project Maintenance and Management Concepts**	**8 hours**

Software Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Need of Maintenance. Project management concepts, Planning the software project, Estimation: Software Measurement and Metrics, Various Size Oriented.

Measures-LOC based, FP based, Halestead's Software Science, Cyclomatic Complexity Measures: Control Flow Graphs, Use-case based empirical estimation COCOMO- A Heuristic estimation technique, staffing level estimation, team structures, risk analysis and management. Configuration Management, Software reengineering, reverse engineering, restructuring forward engineering, Clean Room software engineering. Case Tools.

Total Lecture Hours	**45 hours**

Textbook:

S.No	**Book Title**	**Author**
1	Software Engineering	KK Aggarwal and Yogesh Singh
2	Software Engineering: A Practitioner's Approach	RS Pressman
3	Fundamentals of Software Engineering	Rajib Mall

Reference Books:

S.No	**Book Title**	
1	Software Engineering	Pankaj Jalote
2	Fundamentals of Software Engineering	Ghezzi, M. Jarayeri, D. Manodrioli
3	Software Engineering	Kassem Saleh
4	Software Engineering	Ian Sommerville

Self-Study Contents Links:		**Duration**
Moocs	https://onlinecourses.nptel.ac.in/e-learning/preview/noc20_cs68	18(hr)
Video Lectures	https://www.youtube.com/playlist?list=PLh11ucJN276JGE-oNKj2DDcBQmRKV_SQB	13(hr)
Video Lectures	https://www.youtube.com/watch?v=NlLM3sVF8wY&t=20969s	6(hr)
Video Lectures	https://www.youtube.com/playlist?list=PLZ2ps 7DhBYt5yvXrYAjjWtf5 O399Xea	17(hr)

Course Code: BCSDS0611	Course Name: Applied Data Science and AI	L	T	P	C
Course Offered in: CSE (Data Science)		3	0	0	3

Pre-requisite: Basic Knowledge of Python and R

Course Objectives: To acquire knowledge of autoencoders, GANs, and TensorFlow-based deep learning techniques. To build knowledge of R programming, data analysis, and GUI development techniques. To learn modern data engineering solutions using databases and vector databases. Gain understanding of LLMs, GPT architecture, transformers, and Lang Chain applications. To build prompt engineering and Generative AI applications using modern APIs and frameworks.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Describe the concepts of autoencoders, GANs, and TensorFlow-based deep learning techniques.	K2
CO2	Apply R programming and data analysis techniques for visualization and GUI development.	K3
CO3	Describe database connectivity, vector databases, and similarity search techniques used in data engineering.	K2
CO4	Classify the concepts of LLMs, GPT architecture, transformers, and Lang Chain applications.	K2
CO5	Apply prompt engineering and Generative AI APIs to develop basic AI applications.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	-	-	-	1	1	1	3	2	1
CO2	3	3	2	2	3	-	-	-	1	1	1	3	3	1
CO3	3	2	2	1	3	-	-	-	1	1	2	3	3	1
CO4	3	2	1	1	2	1	-	-	1	1	1	3	2	2
CO5	3	3	3	2	3	1	-	-	2	2	1	3	3	3

Course Contents / Syllabus

Unit 1	Autoencoders and Generative Adversarial Networks (GANs)	7 hours

Introduction to Deep Learning and TensorFlow, TensorFlow Functions and Computational Graphs, Customizing Models and Training Algorithms, TF Transform and Data Pipelines, Efficient Data Representations, Principal Component Analysis (PCA) using Linear Autoencoders, Undercomplete and Stacked Autoencoders, Unsupervised Pre-training using Stacked Autoencoders, Denoising Autoencoders, Sparse Autoencoders, Variational Autoencoders (VAEs), Introduction to Generative Models, Generative Adversarial Networks (GANs), GAN Architecture and Applications.

Unit 2	R Programming and Graphical User Interfaces	6 hours

Introduction to R Programming, Built-in Functions in R, Data Types and Data Structures, Vectors, Matrices, Arrays, Lists and Data Frames, Data Import and Export Techniques, Data Manipulation using Dplyr and Stringr Packages, Descriptive Statistics and Exploratory Data Analysis (EDA), Data Visualization in R, Building and Managing R Packages, Flex dashboard for Interactive Dashboards, GUI Development using R-Shiny.

Unit 3	Data Engineering and Vector Databases	10 hours
Introduction to Data Engineering, Database Connectivity using Python and SQLite, Executing DDL and DML Queries using Python, Data Processing and Query Handling, Introduction to Vector Databases, Architecture of Vector Databases, Embeddings and Vector Representations, Indexing Techniques in Vector Databases, Distance Metrics and Similarity Measures, Nearest Neighbor Search Algorithms, Introduction to Open-Source Vector Databases, Working with Chroma and Milvus Databases, Applications of Vector Databases in AI Systems.		
Unit 4	**Large Language Models (LLMs) and GPT**	**11 hours**
Introduction to Natural Language Processing and Evolution of LLMs, Working of Large Language Models, Tokenization and Next Word Prediction, Training and Fine-tuning of LLMs, Text Generation and Embeddings, LLM Embeddings and Semantic Search, Sentiment Analysis using LLMs, Applications of LLMs across Industries, Introduction to LangChain Framework, Components of LangChain: Prompts, Chains, Retrievers, Memory, Parsers and Agents, Hugging Face Ecosystem and APIs, Attention Mechanism and Transformer Architecture, GPT Architecture and Training Process, Pre-training and Fine-tuning of GPT Models, Building Basic GPT Models using TensorFlow, Introduction to OpenAI and ChatGPT, ChatGPT Architecture and Applications.		
Unit 5	**Prompt Engineering and Generative AI APIs**	**11 hours**
Introduction to Generative AI, Fundamentals of Prompt Engineering, Prompt Design Techniques and Best Practices, Zero-shot, One-shot and Few-shot Prompting, Hands-on with Generative AI APIs, Working with GPT APIs, DALL·E for Image Generation, Whisper for Speech Processing, Midjourney for AI Art Generation, Working with Google Gemini APIs, Introduction to Meta LLaMA Models, Hugging Face Models and Transformers, Building Generative AI Applications using Lang Chain, Introduction to Retrieval-Augmented Generation (RAG), Ethical Considerations and Future Trends in Generative AI.		
	Total Lecture Hours	**45 Hours**

Textbook:

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, MIT Press.
2. Bert Gollnick, Generative AI with Python: The Developer's Guide to Pretrained LLMs, Vector Databases, Retrieval-Augmented Generation, and Agentic Systems, Rheinwerk Publishing.

Reference Books:

3. Ben Auffarth, Generative AI with LangChain, Packt Publishing.
4. Lewis Tunstall, Leandro von Werra, and Thomas Wolf, Natural Language Processing with Transformers, O'Reilly Media.

NPTEL/ YouTube/ Faculty Video Link:
Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	1. https://shorturl.at/5BxHw 2. https://shorturl.at/BssXy	60 Hrs
Video Lectures	3. https://www.youtube.com/@AndrejKarpathy 4. https://www.youtube.com/@TensorFlow	80 Hrs
Web References	1. https://docs.langchain.com/oss/python/deepagents/overview 2. https://www.r-project.org/about.html	40 Hrs

Course Code: BCSE0614	Course Name: WEB DEVELOPMENT USING MEAN STACK	L	T	P	C
Course Offered in: CSE/CS/CSR-R/M.TECH(INT) /CSE(DS)/CSE(CS)		3	0	0	3

Pre-requisite: Basic knowledge of HTML, CSS and ES6 required.

Course Objectives: This course focuses on how to design and build static as well as dynamic webpages and interactive web applications. Students examine advanced topics like Angular, nodejs, Mongodb and Express framework for interactive web applications that use rich user interfaces.

Course Outcome: After completion of the course, the student will be able to

S.No	Course Outcome	Bloom's Level
1	Explain, analyze and apply the role of server-side scripting language like Nodejs in the workings of the web and web applications.	K3
2	Demonstrate web application framework i.e., Express is to design and implement typical dynamic web pages and interactive web based applications.	K4
3	Apply the knowledge of Typescript that are vital in understanding angular is, and analyze the concepts, principles and methods in current client-side technology to implement angular application over the web.	K3
4	Analyze build and develop single page application using client-side programming i.e. angular js and also develop a static web application.	K4
5	Understand the impact of web designing by database connectivity with Mongodb in the current market place where everyone use to prefer electronic medium for shoping, commerce, and even social life also.	K3

CO-PO Mapping (Scale: 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	3	1	3	-	-	1	3	-	3	2	2	3
CO2	2	2	3	1	3	-	-	1	-	-	2	2	2	3
CO3	2	2	2	2	2	-	-	-	2	-	2	2	2	2
CO4	2	1	2	1	2	-	-	-	1	1	2	2	1	2
CO5	2	1	3	2	3	-	-	-	3	2	3	2	1	3

Course Contents / Syllabus

Unit 1	Introduction to Nodejs	8 hours

Installing Nodejs, Node in-built packages (buffer, fs, http, os, path, util, url) Node.js Units, File System Unit, Json data, Http Server and Client, Error handling with appropriate HTTP, Callback function, asynchronous programing REST API's(GET, POST PUT, DELETE UPDATE), GraphQL, Promises, Promise Chaining, Introduction to template engine (EJS).

Unit 2	Express Framework	8 hours

Configuring Express, Postman configuration, , Environment Variables, Routing, Defining pug templates, HTTP method of Express, URL binding, middleware function, Serving static files, Express sessions, REST full API's, FORM data in Express, document modeling with Mongoose.

Unit 3	Basics of Angular js	8 hours

Typescript, Setup and installation, Power of Types, Functions, Function as types Optional and default parameters, Arrow functions, Function overloading, Access modifiers, Getters and setters, Read-only & static, Abstract classes, Interfaces, Extending and Implementing Interface, Import and Export Units.

Unit 4	Building Single Page App with Angular js	8 hours

MVC Architecture, One-way and Two-way data binding, AngularJS Expressions, AngularJS Controllers, AngularJS Units, adding controller to a Unit, Component, Dependency Injection, Filters, Tables, AngularJS Forms and Forms validation, Select using ng-option, AngularJS AJAX.

Unit 5	Connecting Angular js with MongoDB	8 hours

Environment Setup of Mongodb, data modeling, The current SQL/NoSQL landscape, Create collection in Mongodb, CRUD Operations in MongoDB. Mongo's feature set, Introduction to Mongoose, Understanding mongoose schemas and datatypes, Connecting Angular with mongoDB using API.

Total Lecture Hours	**40 hours**

Text Books

S.NO	Book Title	Author
1.	"Web Application Development with MEAN", 3rd Illustrated Edition 2017, Packt Publications.	Amos Q. Haviv , Adrian Mejia , Robert Onodi
2.	"Getting MEAN with Mongo, Express, Angular, and Node", 2nd Edition 2016, Addison Wesley Publication.	Simon Holmes , Clive Herber
3.	"Comprehensive guide to learn Node.js", 1st Edition, 2018 BPB Publications.	Dhruti Shah
4.	"Learning Angular",3rd Edition,2017 Packt publications	Christoffer Noring, Pablo Deeleman

Reference Books:

S.NO	Book Title	Author
1.	"Fullstack Angular: The Complete Guide to AngularJS and Friends", 4th edition, 2020 International Publishing.	Anthony Accomazzo, Ari Lerner, and Nate Murray
2.	"Full-Stack Angular, Type Script, and Node: Build cloud-ready web applications using Angular 10 with Hooks and GraphQL", 2nd edition, 2017 Packt Publishing Limited.	David Cho
3.	"Complete node.js: The fast guide: Learn complete backend development with node.js"5th edition, 2017 SMV publication.	Richard Haltman & Shubham Vernekar
4.	"Mastering Node.js: Build robust and scalable real-time server-side web applications efficiently" 2nd edition,2016 Packt Publishing Limited.	Glenn Geenen, Sandro Pasquali , Kevin Faaborg
5.	Express & MongoDB Development ,kindle edition,2019 international publishing.	Greg Lim
6.	"AngularJS Master Angular.js with simple steps, guide and instructions" 3rd edition, 2015 SMV publication.	Daniel Perkins
7.	"MongoDB Basics", 2nd edition ,2018 International Publication.	Peter Membrey, David Hows, Eelco Plugge

NPTEL/ YouTube/ Faculty Video Link:

Unit 1	https://youtu.be/BLl32FvcdVM https://youtu.be/fCACk9ziarQ https://youtu.be/YSyFSnisip0 https://youtu.be/mGVFltBxLKU https://youtu.be/bWaucYA1YRI
Unit 2	https://youtu.be/7H_QH9nipNs https://youtu.be/AX1AP83CuK4 https://youtu.be/SccSCuHhOw0 https://youtu.be/lY6icfhap2o https://youtu.be/z7ikpQCWbtQ
Unit 3	https://youtu.be/0LhBvp8qpro https://youtu.be/k5E2AVpwsko https://youtu.be/SQJkj0WYWOE.list=PLvQjNLQMdagP3OzoBMfBT48uJ-SPfSsWj https://youtu.be/0eWrpsCLMJQ.list=PLC3y8-rFHvwhBRAgFinJR8KHIrCdTkZcZ https://youtu.be/ZSB4JcLLrIo
Unit 4	https://youtu.be/0LhBvp8qpro https://youtu.be/k5E2AVpwsko https://youtu.be/SQJkj0WYWOE.list=PLvQjNLQMdagP3OzoBMfBT48uJ-SPfSsWj https://youtu.be/0eWrpsCLMJQ.list=PLC3y8-rFHvwhBRAgFinJR8KHIrCdTkZcZ https://youtu.be/ZSB4JcLLrIo

Unit 5	https://youtu.be/Kvb0cHWFkdc https://youtu.be/pQcV5CMara8 https://youtu.be/c3Hz1qUUIyQ
	https://youtu.be/Mfp94RjugWQ https://youtu.be/SyEQLbbSTWg

Mode of Evaluation

CIE					ESE	Total
ST1	**ST3**	**TA 15**	**TA 25**	**Attendance 5**		
35		**15**			**100**	**150**

Course Code: BCSDS0612	Course Name: Natural Language Processing and Social Media Analytics	L	T	P	C
Course Offered in: CSE (Data Science)		3	0	0	3

Prerequisite: Python, R, Basic Statistics.

Course Objectives: To understand the fundamentals of social media analytics and social network analysis. To apply web analytics tools and techniques for analyzing social media data and campaigns. To analyze social media platform data using visualization and network analysis techniques. To apply natural language processing techniques for text mining and sentiment analysis. To evaluate user behavior, information diffusion, and ethical issues in social media analytics applications.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Describe the concepts of social media analytics, social networks, and web analytics techniques.	K2
CO2	Apply web analytics tools and social media metrics to analyze online campaigns and user interactions.	K3
CO3	Analyze social media data using network analysis, visualization, and platform-specific analytics techniques.	K4
CO4	Apply natural language processing techniques for text preprocessing, classification, and sentiment analysis.	K3
CO5	Analyze user behavior, information diffusion, and ethical issues in real-world social media analytics applications.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	-	-	-	1	1	1	3	2	1
CO2	3	3	2	2	3	-	-	-	1	1	1	3	3	1
CO3	3	3	3	2	3	1	-	-	1	1	2	3	3	2
CO4	3	3	2	2	3	-	-	-	1	1	2	3	3	2
CO5	3	3	2	3	2	2	1	1	2	2	2	3	2	3

Course Contents / Syllabus

Unit 1	**Introduction to Social Media Analytics**	**08 hours**

Introduction to Social Media Analytics (SMA), Evolution and Importance of Social Media Analytics, Social Media Landscape and Platforms, Applications of SMA in Business, Healthcare, Marketing, Education, and Governance, SMA in Small and Large Organizations, Social Media Data Sources and Characteristics, Introduction to Social Networks, Nodes, Edges, Ties and Influencers, Social Network Models and Web Data, Graphs and Matrices for Social Networks, Basic Network Measures: Degree, Centrality, Density and Connectivity, Introduction to Information Visualization in Social Media Analytics.

Unit 2	**Web Analytics Tools and Techniques**	**08 hours**

Introduction to Web Analytics, Clickstream Analysis and User Navigation Patterns, A/B Testing and Online Surveys, Key Performance Indicators (KPIs) in Social Media, Google Analytics and Web Traffic Analysis, Social Media Metrics and Performance Evaluation , Web Crawling and Indexing Techniques, Introduction to Micro-text Analysis, Natural Language Processing Techniques for Social Media Data, Link Analysis and Network Evolution, Affiliation and Identity in Social Networks, Social Campaign Analysis and

Evaluation.

Unit 3	**Social Media Platforms and Data Analytics**	**09 hours**

Introduction to Analytics in Social Media Platforms, Facebook Analytics: Audience Insights, Reach and Engagement Analysis, Post-performance Analysis and Campaign Monitoring, Facebook Business Manager, Analytics for LinkedIn, Instagram, YouTube, and X, Social Media Campaign Design and Outcome Evaluation, Network Analysis and Influence Maximization, Link Prediction and Collective Classification, Processing and Visualization of Social Media Data, Applications in Advertising, Marketing Analytics, and Game Analytics using tools such as PyCharm and Unity Technologies.

Unit 4	**Natural Language Processing and Text Analytics**	**10 hours**

Introduction to Natural Language Processing (NLP), History and Applications of NLP, NLP Libraries in Python: NLTK, spaCy and Gensim, Text Preprocessing Techniques: Tokenization, Stemming, Lemmatization and Stop-word Removal, Parts of Speech Tagging and Named Entity Recognition, Topic Modelling and Text Representation Techniques, One Hot Encoding, Bag of Words (BoW), N-Grams and TF-IDF, Text Classification using Naive Bayes, Logistic Regression, Decision Trees and Support Vector Machines (SVM), Introduction to Deep Learning Models for NLP: CNN, RNN, LSTM and Transformer-based Models.

Unit 5	**Advanced Social Media Analytics and Applications**	**10 hours**

Clustering of Short Texts and User-generated Content, Sentiment Analysis and Opinion Mining, Fake News Detection and Spam Filtering, User Behavior Analysis and Engagement Modelling, Bot Detection and Demographic Inference, Social Network Analysis: Community Detection, Clustering and Information Diffusion, Visualization Techniques: Word Clouds, Sentiment Graphs and Network Diagrams, Tools for Social Media Analytics: NLTK, spaCy, Gensim, NetworkX, Gephi and Tweepy, Ethical Issues in Social Media Analytics: Privacy, Bias and Misinformation, Case Studies on Real-world Social Media Data Analysis such as Election Sentiment Tracking, Crisis Monitoring during Pandemics, and Misinformation Spread Analysis.

Total Lecture Hours: 45 Hours

Textbook:

1. **Mathew Ganis and Avinash Kohirkar, Social Media Analytics, IBM Press.**
2. **Jim Sterne, Social Media Metrics: How to Measure and Optimize Your Marketing Investment, Wiley.**
3. **Matthew A. Russell and Mikhail Klassen, Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Instagram, GitHub, and More, O'Reilly Media.**
4. **Akshay Kulkarni and Adarsha Shivananda, Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python, A press.**

Reference Books:

1. **Bing Liu, Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers.**
2. **Christopher D. Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT Press.**

NPTEL/ YouTube/ Faculty Video Link:

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	1. https://shorturl.at/098Mx 2. https://shorturl.at/abeVv	60 Hrs
Video Lectures	1. https://www.youtube.com/@krishnaik06 2. https://www.youtube.com/@freecodecamp	80 Hrs
Web References	3. https://networkx.org/documentation/stable/tutorial.html 4. https://towardsdatascience.com/latest/	20 Hrs

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

Course Code: BCSE0615	Course Name: Web Development using MERN STACK with Devops	L	T	P	C
Course Offered in: CSE/CSE(R)/IT/M.Tech Int. (CSE)/ CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		3	0		3

Pre-requisite: Basic knowledge of HTML, CSS, JavaScript, and fundamental programming concepts in any language.

Course Objectives: This course aims to equip students with practical skills in building dynamic full-stack applications using React, Node.js, Express, and MongoDB. It also introduces DevOps principles, Agile methodologies, and CI/CD automation using tools like GitHub, Jenkins, Docker, and Kubernetes.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Apply the knowledge of ES6 that are vital to implement react application over the web.	K3
CO2	Implement and understand the impact of web designing by database connectivity with MongoDB.	K3
CO3	Explain, analyze and apply the role of server-side scripting language like Nodejs and Express js framework.	K4
CO4	Apply the benefits of DevOps over other software development processes to Gain insights into the DevOps environment.	K4
CO5	Demonstrate popular open-source tools with features and associated terminology used to perform Continuous Integration and Continuous Delivery.	K3

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	1	3	1	1	-	1	-	2	2	2	3
CO2	3	2	3	1	3	1	1	-	1	-	2	2	2	3
CO3	3	2	3	1	3	1	1	2	1	2	2	2	2	2
CO4	3	1	1	1	3	2	-	-	2	1	2	2	1	2
CO5	3	1	1	1	3	1	-	-	2	1	2	2	1	3

Course Contents / Syllabus

Unit 1	Modern Frontend Development with React JS	9 hours

Introduction: to React, Setting up React projects, React Folder Structure, JSX, Functional Components, and React Hooks (useState, useEffect, use Context), Props vs State, Stateless vs Stateful Components, Component Lifecycle, Routing with react-router-dom Introduction to Next.js for Server-Side Rendering and SEO, Integrating APIs with axios or fetch,.

Unit 2	UI Libraries & MongoDB Integration	10 hours

UI Libraries : Introduction to Google Material UI, Tailwind CSS, Theming and responsive design with MUI, Creating reusable UI components (Navbar, Toolbar, Dialogs), Introduction to MongoDB, CRUD Operations using Mongoose Indexes: Creating, Getting, Dropping, Aggregation Pipelines and Query Optimization, Data Modeling & Relationships, Auto-Sharding and Replication Concepts, Modern MongoDB tools: MongoDB Atlas, Compass.

Unit 3	Backend with Node.js & Express	10 hours

Node.js: Introduction to Node.js Setting up Express applications, Routing & Middleware, Restful API Design , JWT-based Authentication & Authorization ,Working with dotenv, CORS, body-parser. Express MVC Architecture in Express, File Upload Handling with Multer, Connecting Frontend with Backend via APIs, Introduction to GraphQL.

Unit 4	DevOps Culture and Lifecycle	8 hours

Introduction to DevOps: Culture & Principles, DevOps Lifecycle and Agile vs DevOps, SDLC Models: Agile, Lean, Scrum, Kanban, Scrum: Process flow, Planning, Testing, Release Management, Introduction to Site Reliability Engineering (SRE), Overview of Terraform/Ansible, Configuration Management basics

Unit 5	CI/CD, Docker & Kubernetes	8 hours

Git & GitHub: Introduction to Git, GitHub, Branching strategies, CI/CD Pipelines: Jenkins, GitHub Actions, Creating Jenkins Jobs and Pipelines, Integrating Git, Maven, and SonarQube, Introduction to Docker: Images, Containers, Lifecycle ,Building custom Docker Images, Docker Compose for multi-container apps, Container orchestration with Kubernetes
Creating Deployments, Services, and scaling with K8s, Basic monitoring tools: Prometheus and Grafana

Total Lecture Hours	45 hours

Textbook :		
S.No	**Book Title**	**Author**
1.	Hands-On MERN Stack Web Development: Build production-grade web apps and APIs with MERN stack ,3rd Edition (2019)	Shama Hoque
2.	Beginning MERN Stack: Build and Deploy a Full Stack MongoDB, Express, React, Node.js App 2nd Edition (2021)	Greg Lim

Reference Books:		
S.No	**Book Title**	**Author**
1.	The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations 2nd Edition (2003).	Gene Kim, Jez Humble, Patrick Debois, John Willis
2.	Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, 2nd Edition (2004).	Jez Humble, David Farley
3.	Test and Deploy Production-Grade Web Applications 1st Edition (2023).	Nabendu Biswas

NPTEL/ Youtube/ Faculty Video Link:	
Unit 1	https://youtu.be/CvCiNeLnZ00
Unit 2	https://youtu.be/4nKWREmCvsE
Unit 3	https://youtu.be/f2EqECiTBL8
Unit 4	https://youtu.be/6GQRb4fGvtk
Unit 5	https://youtu.be/kTp5xUtcalw

LAB Course Code: BCSDS0653	LAB Course Name: BIG DATA ANALYTICS LAB	L	T	P	C
Course Offered in: CSE (Data Science)		0	0	4	2

Pre-requisite: Basic knowledge of Programming, DBMS, Operating Systems, Linux environment, and fundamental concepts of Big Data and Distributed Computing.

Course Objectives:

This course provides hands-on experience in installing, configuring, and managing the Hadoop ecosystem, developing practical skills in Linux commands, HDFS file management, and MapReduce programming, enabling students to design data processing applications using Apache Pig, Hive, and HBase, familiarizing them with Apache Sqoop and Flume for big data integration, and introducing real-time analytics using Apache Spark and Scala for scalable big data applications.

Course Outcome: After completion of the course, the student will be able to	**Bloom's Knowledge Level (KL)**
CO1 Apply Hadoop, Linux, and HDFS to configure the environment and perform data management operations in a distributed computing environment.	K3
CO2 Apply Hadoop ecosystem tools including MapReduce, Pig, Hive, and HBase for big data processing and management.	K3
CO3 Implement data ingestion, integration, and analytics using Sqoop, Flume, Apache Spark, Spark SQL, and Scala.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	3	1	1	1	1	1	1	2	3	1
CO2	3	3	2	2	3	1	1	1	2	2	1	3	3	2
CO3	3	3	3	3	3	1	1	1	2	2	2	3	3	3

List Of Practical's

S. No.	Name of Experiments/ Programs	CO
1.	Installation of VMWare to set up the Hadoop environment and its ecosystems.	CO1
2.	Install and configure Apache Hadoop in standalone and pseudo-distributed modes and verify cluster services using Hadoop web-based monitoring tools.	CO1
3.	Implementing the basic commands of LINUX Operating System – File/Directory creation, deletion, update operations.	CO1
4.	Perform HDFS operations: upload, download, copy, move, delete, view content, and file size analysis.	CO1
5.	Implement the Word Count Map Reduce program to understand Map Reduce Paradigm.	CO2
6.	Implement matrix multiplication using Hadoop MapReduce.	CO2
7.	Write Pig Latin scripts using load, filter, group, join, project, and sort operations	CO2
8.	Use Hive for database, table, view, function, and index management with SQL-like queries.	CO2
9.	Install Hbase and perform CRUD operations using Hbase Shell.	CO2
10.	Import/export data between MySQL and Hadoop using Apache Sqoop.	CO3
11.	To collect streaming data using Netcat and store it into HDFS using Apache Flume.	CO3
12.	Implement Spark Core Processing RDD to run the Word Count program.	CO3
13.	Implement Spark Core Processing RDD to read a table stored in a database and calculate the number of people for every age	CO3
14.	Implement Spark SQL to analyze a dataset and perform aggregate operations (COUNT, SUM, AVG, GROUP BY).	CO3
15.	Implement data manipulation using Scala collections (map, filter, reduce)	CO3
		Total Hours: 30 hrs.

LAB Course Code: BCSE0653Y	LAB Course Name: Software Engineering and Design Lab	L	T	P	C
Course Offered in: CSE (Data Science)		0	0	4	2

Pre-requisite: Basic knowledge of computer fundamentals and software development processes.

Course Objectives:

The course develops students' skills in software development across different phases of the Software Development Life Cycle (SDLC). It emphasizes software quality assurance, software development models, UML-based design, and test case generation. Students gain practical knowledge of structural, behavioral, and architectural modeling using StarUML along with software testing
techniques and test suite design.

Course Outcome: After completion of the course, the student will be able to

CO1	Apply Software Engineering principles and appropriate software development life cycle models in software projects to develop IEEE standard SRS documents.	K3
CO2	Design and represent software projects using different UML diagrams based on software design concepts.	K3
CO3	Perform software testing using various testing techniques and generate test cases to ensure software quality.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	-	2	-	2	-	-	1	2	2	-	2
CO2	2	2	3	-	2	-	-	-	-	-	-	1	2	2
CO3	2	3	2	2	3	-	-	-	-	1	2	3	2	-

List Of Practical's (Indicative & Not Limited To)

S.No	Experiment (Case Study Based Question)	CO Mapping
1	Team formation and allotment of Mini project: Problem statement, Literature survey, Requirement. Analysis. OR Case Study : (Airline/Railway Reservation System, College ERP System, Banking System, Food Delivery Application, ATM Management System, Online Ticket Booking System, Mobile Recharge Application, Vehicle Rental System etc.	CO1
2	To draw the Use Case Diagram for the selected mini project or case study identified in Experiment 1 using UML modelling techniques	CO2
3	To draw the Data Flow Diagram (DFD) at different levels for the mini project or case study identified in Experiment 1	CO2
4	To design an Entity Relationship (ER) Diagram with multiplicity for the mini project or case study identified in Experiment 1.	CO2
5	To prepare a Software Requirement Specification (SRS) document based on IEEE recommended standards for the mini project or case study identified in Experiment 1.	CO2
6	To draw the State Chart Diagram for the mini project or case study identified in Experiment 1 using UML modelling techniques.	CO3
7	To design Object and Class Diagrams for the mini project or case study identified in Experiment 1.	CO3
8	To create a Sequence Diagram for Software Design Description (SDD) of the mini project or case study identified in Experiment 1	CO3

9	To create a Collaboration Diagram for Software Design Description (SDD) of the mini project or case study identified in Experiment 1	CO3
10	To draw the Activity Diagram for the mini project or case study identified in Experiment 1	CO3
11	To create the Component Diagram for the mini project or case study identified in Experiment 1	CO3
12	To design the Deployment Diagram for the mini project or case study identified in Experiment 1	CO3
13	Create a test plan document for any application (e.g., Library Management System, identified in Experiment 1)	CO4
14	To Create Test Scenario for any application(e.g. Google pay login Unit)	CO4
15	To Design test cases for Gmail inbox, Google Search page.	CO4
16	To design test cases for a banking application or Login System using the Boundary Value Analysis (BVA) testing technique.	CO4
17	To Design Test Cases for Social Media Application Using Regression Testing.	CO4
18	To Design Test Cases Using Equivalence Class Partitioning for Login Authentication System.	CO4
19	To design test cases for an Online Payment Gateway using the Decision Table testing technique.	CO4
20	Create test cases for a program which determine whether an integer is prime or not by using path testing.	CO4
21	Create test cases for a program which determine whether an integer is prime or not by using Cyclomatic complexity.	CO4
22	To generate independent execution paths and test cases for a factorial program using Cyclomatic Complexity	CO4
23	Consider the code to arrange the nos. in ascending order. Generate the test cases for loop coverage and path testing. Check the adequacy of the test cases through mutation testing and compute the mutation score for each.	CO4
24	To Estimation of Test Coverage Metrics and Structural Complexity.	CO4
25	Perform Analysis and comparisons of at least 3 testing tools (eg. Qtp,winrunner, postman)	CO4
26	Perform Analysis and comparisons bug tracking tool (e.g., Bugzilla, Bug bit)	CO4
27	Perform Analysis of any web testing tool (e.g., Selenium)	CO4
28	Perform Analysis of any open source-Testing tool (e.g., Test link, Test Rail)	CO4
29	To estimate software size, development effort, development time, and project cost using LOC (Lines of Code) based estimation techniques.(refer Experiment-1)	CO5
30	To estimate the effort, development time, staffing requirement, and cost of a mini project or Case study identified in experiment 1 using the Basic COCOMO Model.	CO5
		Total Hours: 60 hrs.

LAB Course Code: BCSDS0611P	LAB Course Name: Applied Data Science and AI Lab	L	T	P	C
Course Offered in: CSE/CS(DS)CSE(AI)/CSE(AIML)		0	0	2	1

Pre-requisite:

Course Objectives: To understand and apply machine learning and deep learning techniques for solving real-world problems. To use R, SQL, and Python for data manipulation and database operations. To build intelligent AI applications using transformers, vector databases, and Generative AI tools..

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply machine learning and deep learning techniques (Autoencoders, GANs) for data analysis and image generation.	K3
CO2	Apply R, SQL, and Python for data manipulation and database operations.	K3
CO3	Analyze and develop AI applications using transformers, vector databases, and generative AI tools.	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	1	1	1	2	–	–	1	–	–	–	3	2	1
CO2	1	1	–	–	2	–	–	–	1	–	–	2	2	2
CO3	1	2	2	1	2	1	–	1	1	1	1	3	3	2

List of Practicals (Indicative & Not Limited To)	CO Mapping
1. Implement Principal Component Analysis (PCA) using Autoencoder in TensorFlow.	CO1
2. Develop a Denoising Autoencoder for image reconstruction.	CO1
3. Implement a Generative Adversarial Network (GAN) for handwritten digit generation.	CO1
4. Image Generation using Generative AI APIs (e.g., DALL·E).	CO1
5. Perform data manipulation using R (dplyr, stringr packages).	CO2
6. Conduct Exploratory Data Analysis (EDA) using R.	CO2
7. Develop an interactive dashboard using R Shiny.	CO2
8. Implement Database Connectivity, Table Creation, and Data Insertion using MySQL and Python (PyMySQL/Connector).	CO2
9. Implement Data Retrieval and Filtering using SQL Queries in Python	CO2
10. Implement a Simple Document Search System using Vector Database (FAISS or Similar Library).	CO3
11. Implement Text Classification using Pretrained Transformer Models.	CO3
12. Perform Sentiment Analysis using Hugging Face Transformers	CO3
13. Develop a Simple LangChain-based Application.	CO3
14. Implement Text Generation using GPT API.	CO3
15. Design a Question-Answering System using Prompt Engineering Techniques	CO3
	Total Hours: 30 hrs.

LAB Course Code: BCSE0614P	LAB Course Name: WEB DEVELOPMENT USING MEAN STACK	L	T	P	C
Course Offered in: CSE/CS(Ds)/IT//CSE(AI)/CSE(AIML)/CSE(DS)/CSE(CS)		0	0	2	1

Pre-requisite: Basic knowledge of HTML, CSS and ES6 required.

Course Objectives: This course focuses on how to design and build static as well as dynamic webpages and interactive web applications. Students examine advanced topics like Angular, NodeJS, MongoDB and Express framework for interactive web applications that use rich user interfaces.

Course Outcome: After completion of the course, the student will be able to

CO1	Develop server-side web applications using Node.js and Express framework by implementing Units, asynchronous programming, REST APIs, middleware, template engines, and document modeling with Mongoose.	**K6**
CO2	Design and build dynamic Single Page Applications (SPAs) using AngularJS and TypeScript by applying components, Units, data binding, controllers, dependency injection, forms, and AJAX concepts.	**K6**
CO3	Create and manage MongoDB databases by performing data modeling, CRUD operations, schema design using Mongoose, and integrating Angular applications with MongoDB through APIs.	**K6**

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	3	2	3	–	–	–	1	1	–	1	3	2
CO2	3	2	3	2	3	–	–	–	2	2	–	1	3	2
CO3	3	3	3	2	3	–	–	–	1	1	–	2	3	2

S.No	List of Practicals
1	Construct a program to demonstrate usage of Node.js fs Unit to read a file.
2	Construct a program to display system information using the os Unit.
3	Construct a program to create and export a simple Node.js Unit.
4	Construct a program to parse a URL using the Node.js url Unit.
5	Construct a program to create a basic HTTP server using Node.js
6	Construct a program to demonstrate callback-based asynchronous execution.
7	Construct a program to illustrate promise creation and resolution.
8	Construct a program to demonstrate promise chaining with multiple operations.
9	Construct a program to show JSON data parsing and stringifying.
10	Construct a program to demonstrate routing in an Express application.
11	Construct a program to demonstrate AngularJS data binding.
12	Construct a program to build a Node.js server that handles GET requests.
13	Construct a program to implement POST request handling using Express.
14	Construct a program to perform CRUD operations using Express and MongoDB.
15	Construct a program to create middleware for logging request details.
16	Construct a program to serve static files using Express.
17	Construct a program to create a MongoDB collection and insert documents.
18	Construct a program to build an AngularJS form with validation.
19	Construct a program to implement an API call using AngularJS AJAX.

20	Construct a program to compare synchronous and asynchronous file operations.
21	Construct a program to demonstrate differences between callbacks and promises.
22	Construct a program to illustrate REST API endpoints with multiple HTTP methods.
23	Construct a program to demonstrate middleware execution order in Express.
24	Construct a program to show one-way and two-way data binding in AngularJS.
25	Construct a program to analyze MongoDB document structure using Mongoose schema.
26	Construct a program to measure execution time of synchronous vs asynchronous code.
27	Construct a program to test REST API endpoints using different HTTP methods.
28	Construct a program to validate user authentication using Express sessions.
29	Construct a program to handle error responses with proper HTTP status codes.
30	Construct a program to demonstrate secure API access using middleware.
31	Construct a program to develop a RESTful API using Node.js, Express, and MongoDB.
32	Construct a program to build a single-page application using AngularJS.
33	Construct a program to integrate AngularJS frontend with Node.js backend.
34	Construct a program to implement user registration and login system.
35	Construct a program to build a full-stack application with CRUD operations and API integration.

Total Hours : 30 Hours

Note: After completing all the laboratory experiments, students are required to obtain a Free Angular Online Course with Certificate - Beginner through ScholarHat

Web Reference	https://www.scholarhat.com/free-course/angular-course?utm_source=chatgpt.com **Free Angular Online Course with Certificate - Beginner Level (8 hours)**

Course Code: BCSE0654A	Course Name: Computer Vision lab	T	P	C
Course Offered in: CSE (Data Science)		0	6	3

Pre-requisite: Basic Knowledge of programming language Python/Advance Python features and Libraries

Course Objectives: To understand the fundamental concepts, techniques, and applications of Computer Vision and Deep Learning..To design and implement image processing and computer vision algorithms using modern tools and frameworks.To develop and train deep learning models for tasks such as image classification, object detection, segmentation, and recognition.To analyze various datasets and improve model accuracy, reliability, and performance through optimization and evaluation techniques.To apply computer vision solutions for real-world problems and generate concise, efficient, and interpretable analytical results.

Course Outcome: After completion of the course, the student will be able to

CO No.	Course Outcome	Bloom's Level
CO1	Solve various Vision and Pattern Association tasks using deep architectures.	K3
CO2	Design appropriate learning rules for each of the deep architectures used in computer vision.	K6
CO3	Deploy training algorithm for pattern association with the help of memory network.	K3
CO4	Design and deploy the models of deep learning with the computer vision use cases.	K6
CO5	Analyze different theories of deep architecture for computer vision	K4

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3
CO1	3	3	2	2	3	-	-	-	-	-	3	3	2
CO2	3	3	3	2	3	-	-	-	-	-	3	3	3
CO3	2	3	2	3	3	-	-	-	-	-	3	3	2
CO4	3	3	3	3	3	1	-	-	1	-	3	3	3
CO5	3	3	2	3	2	-	-	1	-	1	2	3	2

Course Contents / Syllabus

Unit 1	Introduction	14 hours

Introduction to Computer Vision

Introduction to Computer Vision and Applications, Image Representation and Processing, Pixels and Image Fundamentals, Colour Spaces (RGB, HSV, Grayscale, LAB), Image Filtering and Transformations, OpenCV Basics and Image Manipulation, Edge Detection (Sobel, Canny, Laplacian), Feature Descriptors: SIFT, SURF, ORB, Image Categorization and Moving Object Observation, Image Retrieval Based on Content, Morphological Operations, Histogram Equalization and Contrast Enhancement, Thresholding and Noise Removal Techniques, Optical Flow and Background Subtraction Basics

Unit 2	Architectures	14 hours

Representation of Three-Dimensional Moving Scene, CNN Fundamentals and Architectures Overview, LeNet-5 Architecture, AlexNet Architecture, VGGNet and GoogLeNet, ResNet, EfficientNet, MobileNet, RNN Introduction, Convolutional Layers, Pooling, Padding, Stride, Backpropagation in CNNs, Transfer Learning and Pre-trained Models, Batch Normalization and Dropout, Activation Functions and Regularization, Data Augmentation and Fine Tuning, Model Efficiency: Accuracy vs FLOPS vs Parameters

Unit 3	**Image Segmentation & Advanced Vision**	**14 hours**

Popular Image Segmentation Architectures, FCN Architecture, Upsampling Methods, Pixel Transformations and Geometric Operations, Spatial Operations in Image Processing, Instance Segmentation and Localization, Object Detection and Segmentation using CNNs, LSTM and GRU Introduction, Vision Models and Vision Languages, Visual Dialogue and Quality Analysis, Active Contours and Applications, Split & Merge and Mean Shift, Normalized Cuts and Attention Mechanism, U-Net and Mask R-CNN
Overvie

Unit 4	**Object Detection**	**15 hours**

Object Detection and Sliding Windows, R-CNN Architecture, Fast R-CNN and Faster R-CNN, Object Recognition Techniques, 3-D Vision and Geometry, Digital Watermarking, Face Recognition, Instance Recognition and Category Recognition, Objects, Scenes, Activities, Object Classification, YOLO (v5/v8) Architecture, SSD Object Detection, Bounding Box Regression and Non-Maximum Suppression (NMS), Pose Estimation and Human Activity Recognition

Unit 5	**Visualization and Generative Models**	**15 hours**

Benefits of Interpretability, Class Activation Maps, GradCAM and Visualization Techniques, ZFNet, Introduction to Deep Generative Models, GAN Fundamentals, VAE and GAN Combination, Other Variants of VAE and GAN, GAN Improvements, Deep Generative Models Across Domains, Image and Video Applications of GANs, Vision Transformers (ViT), StyleGAN, CycleGAN and Diffusion Models, Explainable AI and Ethical Issues in Generative AI

	Total Lecture Hours	**72 hours**

Textbook:

1. Introductory Techniques for 3D Computer Vision", Edition 2009
2. Szelisk Richard, "Computer Vision: Algorithms and Applications", 2022, The University of Washington Edition, 2022
3. Forsyth D. and Ponce J., "Computer Vision - A Modern Approach", Prentice Hall,, Edition 2015
4. Trucco, E. and Verri, A. (1998). Introductory Techniques for 3D Computer Vision, Prentice- Hall.
5. Davies E. R., "Computer & Machine Vision", Academic Press 4th Edition 2012
6. Simon J. D. Prince, "Computer Vision: Models, Learning, and Inference", Cambridge University Press Edition, 2012

Reference Books:

1. Forsyth D. and Ponce J., "Computer Vision: A Modern Approach", Prentice Hall, 2nd Edition, 2015
2. Prince, Simon J.D. "Computer Vision: Models, Learning, And Inference". Cambridge University Press, 1st Edition, 2012.
3. Ballard D. H., Brown C. M., "Computer Vision", Prentice-Hall, 2008.
4. Craig Alan B., "Understanding Augmented Reality, Concepts and Applications", Morgan Kaufmann, Edition 2013
5. Richard Szeliski, "Computer Vision: Algorithms and Applications (CVAA)", Springer Edition, 2022

Self-Study Content Links:		Duration (in Hrs)
MOOCs	NA	NA
Video Lectures	1. https://nptel.ac.in/courses/106103224	16 hours
	2. https://nptel.ac.in/courses/106102065	33 hrs
	3. https://onlinecourses.nptel.ac.in/noc23_ee78/preview	34 hrs
Web References	**https://opencv.org/university/free-courses/?utm_source=opcvu&utm_medium=menu**	NA

Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.

List Of Practical's (Indicative & Not Limited To)	CO
1. Read and display images using Python and OpenCV 2. Perform image resizing, cropping, and rotation 3. Convert images between RGB, HSV, and Grayscale colour spaces	**CO1**
4. Perform image thresholding using binary and adaptive thresholding 5. Apply image smoothing filters (Gaussian, Median, Bilateral) 6. Implement image sharpening techniques 7. Perform histogram equalization for contrast enhancement 8. Detect edges using Sobel operator 9. Detect edges using Canny edge detector 10. Implement Laplacian edge detection 11. Perform morphological operations: erosion and dilation 12. Perform opening and closing operations on images 13. Detect contours and draw bounding boxes 14. Perform background subtraction in videos 15. Implement optical flow for moving object tracking 16. Extract image features using SIFT 17. Extract image features using SURF 18. Extract image features using ORB 19. Perform image stitching using feature matching 20. Build a simple image retrieval system based on image features	**CO2**
21. Implement image classification using a basic CNN 22. Design and train LeNet-5 on the MNIST dataset 23. Implement AlexNet architecture for image classification 24. Implement VGGNet for image recognition 25. Implement ResNet using TensorFlow or PyTorch 26. Compare CNN architectures based on accuracy and parameters 27. Perform transfer learning using a pre-trained VGG16 model 28. Perform transfer learning using MobileNet 29. Apply data augmentation techniques on image datasets 30. Visualize feature maps of convolutional layers 31. Implement max pooling and average pooling operations 32. Demonstrate padding and stride effects in CNNs 33. Implement backpropagation in a simple neural network	**CO3**
34. Perform semantic segmentation using FCN 35. Implement U-Net for image segmentation 36. Perform instance segmentation using Mask R-CNN 37. Detect objects using Faster R-CNN 38. Implement object detection using YOLOv5 or YOLOv8 39. Perform face detection using Haar Cascade classifier 40. Build a face recognition system using OpenCV 41. Implement object tracking in videos 42. Perform human activity recognition using video sequences	**CO4**
43. Implement attention mechanism in deep learning models 44. Build an image caption generation model using CNN and LSTM 45. Implement GradCAM for CNN interpretability 46. Visualize class activation maps for image classification 47. Generate handwritten digits using GAN 48. Implement image super-resolution using GAN 49. Build a simple Variational Autoencoder (VAE) 50. Implement Vision Transformer (ViT) for image classification	**CO5**
Total Hours	**30 hrs**

LAB Course Code: BCSE0659X	LAB Course Name: Applied Research-based Mini Project	L	T	P	C
Course Offered in: All Engineering Programs		0	0	2	1

Pre-requisite: Basic knowledge of programming, data structures, databases, and fundamental concepts of software development. Familiarity with tools and technologies for system development is desirable.

Course Objectives:

This course aims to develop the ability to design and implement solutions for real-world problems using appropriate tools and technologies. It emphasizes system integration, performance evaluation, and effective documentation, along with promoting research skills through the preparation of a structured research paper.

CO1	Identify and define a domain-specific real-world problem through literature review, objective formulation, and structured project planning suitable for research-based implementation.	**K4**
CO2	Design, develop, and validate a functional prototype/model/system using appropriate tools, technologies, or methodologies while considering performance, ethics, safety, and reliability.	**K6**
CO3	Document, analyze, and present project outcomes through a technical report and a structured research paper in IEEE/Scopus/UGC-approved format, demonstrating research communication and publication readiness.	**K4**

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

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

Unit	Common Applied Research-Based Mini Project Activities (All Branches)	Expected Outcomes
Unit 1: Research Foundation & Project Planning	1. Identify domain-specific problem statement relevant to the branch. 2. Define objectives, hypothesis, and expected outcomes 3. Conduct literature review 4. Select suitable tools, technologies, software, datasets, laboratory methods, or platforms. 5. Design project methodology, workflow, or system/process architecture	Clear research problem formulation, defined objectives, foundational knowledge, and structured project roadmap
Unit 2: Development, Design & Implementation	6. Develop core project components, models, experiments, prototypes, or processes. 7. Integrate relevant Units such as software, hardware, simulation, experimentation, analysis, or field implementation. 8. Apply discipline-specific techniques such as coding, fabrication, experimentation, surveys, analytics, optimization, or modelling.	Functional prototype/model/system/process with interdisciplinary implementation and practical technical skills

	9. Incorporate safety, ethics, reliability, sustainability, and security considerations where applicable	
Unit 3: Testing, Validation & Optimization	10. Perform testing, experimentation, or validation of developed work. 11. Analyze performance, efficiency, feasibility, or effectiveness. 12. Apply analytical, computational, statistical, or intelligent techniques where relevant. 13. Optimize design, methodology, process, or implementation for improved outcomes	Validated and optimized project demonstrating performance improvement, technical analysis, and practical feasibility
Unit 4: Documentation, Research Publication & Presentation	14. Document project design, methodology, implementation, and results. 15. Prepare technical report/project dissertation. 16. Draft a research paper in standard format (Abstract, Introduction, Methodology, Results, Conclusion, References) 17. Format manuscript according to IEEE/Scopus/UGC-approved guidelines. 18. Include proper figures, tables, graphs, citations, and references 19. Present project findings through seminar/poster/presentation. 20. Demonstrate developed system/model/prototype/process. 21. Submit final project report and research paper for evaluation/publication consideration	Complete technical documentation, publishable-quality research paper, professional presentation skills, and final project submission