NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA, GAUTAM BUDDH NAGAR
(An Autonomous Institute)

NIET
Greater Noida
An Autonomous Institute
Estd.2001
Get Future Ready

Affiliated to

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

Evaluation Scheme & Syllabus

For

B.Tech. in Computer Science (CS)

Third Year

(Effective from the Session: 2026-27)

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA

(An Autonomous Institute)

B. TECH (CS)

Evaluation Scheme

SEMESTER V

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	BCSE0502A	Computer Networks	Mandatory	3	1	0	30	20	50		100		150	4
2	BCSML501B	Machine Learning Algorithms	Mandatory	3	1	0	30	20	50		100		150	4
3		Departmental Elective II	Departmental Elective	3	0	0	30	20	50		100		150	3
4		Departmental Elective III	Departmental Elective	3	0	0	30	20	50		100		150	3
5	BCSE0552A	Computer Networks Lab	Mandatory	0	0	4				50		50	100	2
6	BCSML551B	Machine Learning Algorithms Laboratory	Mandatory	0	0	2				25		25	50	1
7		Departmental Elective II Lab.	Departmental Elective	0	0	2				25		25	50	1
8	BCS0555	Cloud Architecture and Infrastructure	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
13		MOOCs (For B.Tech. Hons. Degree)												
		TOTAL		**14**	**2**	**24**	**180**	**120**	**300**	**250**	**400**	**200**	**1150**	**25**

***List of MOOCs Based Recommended Courses for Third year B. Tech Students**

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	BMC0072	Cloud -Native Applications in JAVA	Infosys Wingspan (Infosys Springboard)	14h 49m	1
2.	BMC0068	AWS Certified Solutions Architect Associate 2022	Infosys Wingspan (Infosys Springboard)	7h 49m	0.5
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)

List of Department Elective II & III: -

Sr. No	Subject code	Subject Name	Types of Subjects	Bucket Name	Branch	Sem.
1	BCSAI0512	Advance Java Programming	DE II	Full stack Development	CS	V
2	BCSAI0512P	Advance Java Programming Lab	DE II	Full stack Development	CS	V
3	BCSCY0513	Network Security and Cryptography	DE II	Cyber Security-I	CS	V
4	BCSCY0513P	Network Security and Cryptography Lab	DE II	Cyber Security-I	CS	V
5	BCSAI0517	Programming For Data Analytics	DE II	AI Driven Analytics	CS	V
6	BCSAI0517P	Programming For Data Analytics Lab	DE II	AI Driven Analytics	CS	V
7	BCSE0514	Design Pattern	DE III	Full stack Development	CS	V
8	BCSAI0519N	Business Intelligence and Data Visualization	DE III	AI Driven Analytics	CS	V
9	BCSCY0515	Ethical Hacking	DE III	Cyber Security-I	CS	V

NOIDA INSTITUTE OF ENGINEERING & TECHNOLOGY, GREATER NOIDA

(An Autonomous Institute)

B. TECH (CS)

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	BCSML0601B	Deep Learning Techniques	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	Departmental Elective	3	0	0	30	20	50		100		150	3
4		Departmental Elective V	Departmental Elective	3	0	0	30	20	50		100		150	3
5		Open Elective I	Elective	2	0	0	30	20	50		50		100	2
6	BCSML0651B	Deep Learning Techniques 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 IV Lab	Departmental Elective	0	0	2				25		25	50	1
9	BCS0651	Cloud Application Development	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 and Employability II	Compulsory Audit	0	0	6	60	40	100					NA
12		MOOCs (For B.Tech. Hons. Degree)												
		TOTAL		**14**	**2**	**18**	**150**	**100**	**250**	**225**	**450**	**225**	**1150**	**25**

List of MOOCs Based Recommended Courses for Third year B. Tech Students

S. No.	Subject Code	Course Name	University/ Industry Partner Name	N. of Hours	Credits
1.	BMC0078	Explore Machine Learning using Python	Infosys Wingspan (Infosys Springboard)	17h 7m	1
2.	BMC0134	Kubernetes with HELM: Kubernetes for Absolute Beginners	Infosys Wingspan (Infosys Springboard)	16h 9m	1

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 IV & V: -

Sr. No	Subject code	Subject Name	Types of Subjects	Bucket Name	Branch	Sem.
1	BCSCY0613	Malware Analysis and Reverse Engineering	DE IV	Cyber Security-I	CS	VI
2	BCSCY0613P	Malware Analysis and Reverse Engineering Lab	DE IV	Cyber Security-I	CS	VI
3	BCSML0614	Natural Language Processing and Social Media Analytics	DE IV	AI Driven Analytics	CS	VI
4	BCSML0614P	Natural Language Processing and social media Analytics Lab	DE IV	AI Driven Analytics	CS	VI
5	BCSE0614	Web Development Using Mean Stack	DE IV	Full stack Development	CS	VI
6	BCSE0614P	Web Development Using Mean Stack Lab	DE IV	Full stack Development	CS	VI
7	BCSE0615	Web Development Using Mern Stack with Devops	DE V	Full stack Development	CS	VI
8	BCSCY0612N	Blockchain And Cryptocurrency Security	DE V	Cyber Security-I	CS	VI
9	BCSML0612	Generative AI	DE V	AI Driven Analytics	CS	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(IoT)/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/	-

Course Code: BCSML501B	**Course Name: Machine Learning Algorithms**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech. 3rd year (5th Semester) Computer Science		**3**	**1**	**0**	**4**

Pre-requisite: Basic knowledge of Python programming and Statistics fundamentals.

Course Objectives:

Present the fundamental concepts of machine learning, explain data preprocessing methods, discuss key machine learning algorithms, outline model evaluation and optimization techniques, and provide hands-on experience with programming tools and libraries for solving real-world problems.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO-1	Explain Machine Learning concepts, feature engineering, preprocessing techniques, and PCA for data analysis.	K2
CO-2	Apply regression models, Gradient Descent, Evaluation Metrics, and regularization techniques in Machine Learning	K3
CO-3	Implement classification algorithms for solving binary and multiclass Machine Learning problems effectively.	K3
CO-4	Analyze probabilistic learning methods and ensemble techniques for robust predictive model development	K4
CO-5	Evaluate clustering and anomaly detection algorithms for unsupervised learning and pattern discovery.	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	2	1	3	–	–	2	1	1	–	2	2	2
CO2	2	3	3	–	3	–	–	3	–	1	–	2	2	3
CO3	3	3	3	3	2	1	–	3	–	1	1	2	3	3
CO4	3	3	2	3	2	1	–	3	1	2	–	3	3	3
CO5	3	3	3	3	3	1	1	3	1	2	1	3	3	3

Course Contents / Syllabus

Unit 1 **Introduction to Machine Learning and Feature Engineering** **08 hours**

Introduction to Machine Learning: Motivation, Definition, Applications of Machine Learning
Types of Machine Learning: Supervised, Unsupervised and Reinforcement Learning
Datasets and Features: Features and their types, handing missing data, Dealing with categorical features, Feature Scaling, Feature selection, outlier detection and removal, Principal Component Analysis (PCA) Algorithm

Unit 2 **Regression and Regularization** **09 hours**

Introduction to Regression: Regression Problem**,** Linear regression model (Simple, Multiple and Polynomial) **Errors Measures:** Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) **Cost function:** Formula, intuition and visualization
Gradient Descent: Algorithm, intuition, learning rate and Gradient Descent for linear regression.
Evaluating the Accuracy of Linear Regression Model: R^2 and Adjusted R^2
Regularization: Bias and Variance, Overfitting and Underfitting, L1 and L2 Regularization, Regularized cost function, Regularized Linear Regression

Unit 3 **Classification Algorithms** **11 hours**

Classification: Classification problem, Binary and Multiclass Classification, Decision Boundary
Logistic regression model for binary classification, Cost function of logistic regression model, Regularized logistic regression, One Vs. All strategy for multiclass classification using logistic regression.
Measuring Performance of Classification Model: Accuracy, Precision, Recall, F1-Score, Confusion matrix
Decision Trees: Decision Tree Classification Algorithm, Attribute Selection Measures, Variants of Decision Trees (ID3, C4.5, CART)
Large Margin Classifier: Support Vector Machines
Instance Based Learning: K Nearest Neighbor Algorithm

Unit 4 **Probabilistic Models and Ensembles** **08 hours**

Probabilistic Methods for Classification: Bayesian Learning, Naive Bayes Classifier, Bayesian belief networks **Tree Ensembles:** Using multiple decision trees, Bagging Vs. Boosting, Sampling with replacement, Random Forest Algorithm, Adda boost, Gradient Boosting, Xgboost Algorithm.		
Unit 5	**Clustering Algorithms and Anomaly Detection**	**09 hours**
Unsupervised Learning &Clustering: The problem of clustering, Distance measures **Partition Based Methods:** K-Means Clustering (Intuition, Algorithm, Optimization objective, Initializing K-Means, Choosing number of clusters) K-Medoids and K-Mode clustering algorithms. **Hierarchical Clustering** (Single Linkage, Complete Linkage, Agglomerative Nesting (AGNES), Divisive Analysis (DIANA). **Density-based Clustering:** DBSCAN Algorithm **Anomaly Detection:** Finding unusual events, Gaussian Distribution and Gaussian Mixture Model, Anomaly Detection Algorithm, Developing and evaluating anomaly detection system.		
	Total Lecture Hours	**45 hours**

Textbook:	
Books Details:	**Authors Name**
Machine Learning, McGraw-Hill Education, 1997	Tom M. Mitchell
Machine Learning with Python, Wiley India, 2019	Manaranjan Pradhan & U. Dinesh Kumar
Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd Edition, O'Reilly Media, 2022	Aurélien Géron
Machine Learning with PyTorch and Scikit-Learn, 2nd Edition, Packt Publishing, 2022	Sebastian Raschka, Yuxi Liu, Vahid Mirjalili
Reference Books:	
Books Details:	**Authors Name**
Pattern Recognition and Machine Learning, 1st Edition, Springer, 2006	Christopher M. Bishop
Introduction to Machine Learning, 4th Edition, MIT Press, 2020	Ethem Alpaydin

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	Explore Machine Learning with TensorFlow	14Hrs 09Min
Video Lectures	https://www.youtube.com/watch?v=GwIo3gDZCVQ (Edureka) https://www.youtube.com/watch?v=JxgmHe2NyeY&t=476s https://www.youtube.com/watch?v=53mqteI5TS0 https://www.youtube.com/watch?v=ZftI2fEz0Fw&list=PLKnIA16_Rmvbr7zKYQuBfsVkjoLcJgxHH https://www.youtube.com/watch?v=k9V3DapHEDA	9Hrs 36Min 6Hrs 36Min 23Hrs 26Min 25Hrs 05Hrs 35Min
Web References	https://www.kaggle.com/learn/intro-to-machine-learning https://openlearninglibrary.mit.edu/courses/course-v1:MITx+6.036+1T2019/about https://developers.google.com/machine-learning/crash-course	

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 RequestDispatcher, sendRedirect, Session Tracking: Using cookies, URL rewriting, and HttpSession.

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, Servlet-JSP-JDBC Integration, Login and registration system using JSP and Servlet.

Unit 3	**Spring Framework**	**9 hours**

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

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, Serving HTML pages and static content, Handling HTTP methods, H2 databased Integration, 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, JSON/XML handling, Postman testing and deployment.

	Total Lecture Hours	**45 hours**

Textbook:		
S.No	**Book Title**	
1	B. Basham, K. Sierra, and B. Bates, Head First Servlets and JSP, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2008.	
2	H. Bergsten, Java Server Pages, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2003.	
3	C. Walls, Spring in Action, 6th ed. Shelter Island, NY, USA: Manning Publications, 2022.	
4	C. Walls, Spring Boot in Action, 1st ed. Shelter Island, NY, USA: Manning Publications, 2016.	
5	M. Pollack and O. Gierke, Spring Data: Modern Data Access for Enterprise Java, 1st ed. Sebastopol, CA, USA: O'Reilly Media, 2012.	
Reference Books:		
S.No	**Book Title**	
1	M. Hall and L. Brown, Core Servlets and JavaServer Pages, Volume 1: Core Technologies, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2003.	
2	M. Hall and L. Brown, Core Servlets and JavaServer Pages, Volume 2: Advanced Technologies, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2004.	
3	C. Ho, R. Harrop, and C. Schaefer, Pro Spring 6, 1st ed. New York, NY, USA: Apress, 2023.	
4	M. Keith and M. Schincariol, Pro JPA 2 in Java EE 8, 3rd ed. New York, NY, USA: Apress, 2018.	
Self Study Content Link:		
MOOCs	1. https://nptel.ac.in/courses/106105191 2. https://www.coursera.org/specializations/spring-framework	12 Weeks 50 hrs
Video Lectures	1. https://www.youtube.com/playlist?list=PLlhM4lkb2sEjVsbbZ_kiixY5CcR84IQUg 2. https://www.youtube.com/playlist?list=PL-XjHn7CHrmQhMkVC6KCfmsPHdklvUlQr 3. https://www.youtube.com/playlist?list=PL9ooVrP1hQOEfi91PCFQMawtBJrPpir7y 4. https://www.youtube.com/playlist?list=PL-XjHn7CHrmQhMkVC6KCfmsPHdklvUlQr 5. https://www.youtube.com/playlist?list=PLGRDMO4rOGcNSBOJOlrgQqGpIgo6_VZgR	13 hrs 1 hrs 18 hrs 50 Min 25 hrs
Web References	1. https://spring.io/guides 2. https://docs.oracle.com/javase/tutorial/	

Course Code: BCSCY0513	**Course Name: Network Security and Cryptography**	**L**	**T**	**P**	**C**
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		**3**	**0**	**0**	**3**

Prerequisite: Familiarity Understanding of basic Networking Terms (e.g., IP, LAN, Router), Awareness of Digital Tools (Email, Cloud, Antivirus), Fundamental Logical Thinking, Cybersecurity fundamentals

Course Objectives: This course aims to introduce the fundamental principles and concepts of network security and cryptography, provide insights into common network-based attacks and vulnerabilities and develop the ability to design secure network architectures using modern cryptographic tools and techniques.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the fundamentals of cryptography and its applications in network security.	K1
CO2	Analyze different encryption algorithms and cryptographic protocols.	K2
CO3	Implement secure communication protocols to protect data transmission over networks.	K3
CO4	Identify common network attacks and employ countermeasures to mitigate them.	K3
CO5	Implement secure network architectures using cryptographic techniques.	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	1	-	-	1	-	-	-	-	-	-	2	3	1
CO2	3	3	1	1	2	-	-	-	-	-	-	2	3	2
CO3	2	2	3	2	3	1	-	1	1	1	1	3	2	2
CO4	2	3	2	2	2	2	1	1	-	-	-	2	2	3
CO5	3	2	3	2	3	2	1	1	1	1	2	3	3	3

Course Contents / Syllabus

Unit 1	**Cryptographic Techniques**	**10 hours**

Introduction to Cryptography, Classical Cryptographic Techniques (Caesar Cipher, Monoalphabetic Cipher), Modern Symmetric Encryption Algorithms (AES, DES), Public Key Cryptography and RSA Algorithm (Key Generation, Encryption/Decryption), Digital Signatures and Hash Functions (SHA, Message Authentication Codes)

Unit 2	**Introduction to Network Security**	**9 hours**

Overview of Network Security Principles, Goals and Objectives of Network Security, Threats, Attacks, and Vulnerabilities (e.g., Malware, DoS), Security Policies and Mechanisms (e.g., Access Control, Firewalls), Risk Management and Assessment (e.g., Risk Analysis, Mitigation Strategies).

Unit 3	**Secure Communication Protocols**	**9 hours**

Secure Socket Layer (SSL) / Transport Layer Security (TLS – handshake process, certificate-based authentication), Internet Protocol Security (IPsec – AH/ESP protocols, tunnel and transport mode), Secure Shell (SSH – key-based authentication, port forwarding), Wireless Security Protocols (WPA, WPA2 – PSK, EAP authentication), Virtual Private Networks (VPNs – site-to-site VPN, remote access VPN).

Unit 4	**Network Attacks and Defense Mechanisms**	**8 hours**

Types of Network Attacks (e.g., Denial of Service, Man-in-the-Middle), Intrusion Detection and Prevention Systems (IDS/IPS, Signature-based and Anomaly-based Detection), Secure DNS and DHCP (DNSSEC, DHCP Snooping), Introduction to OSSEC, OSSEC Architecture (Agent-based and Agentless Modes), Event Collection (Syslog, File Monitoring), Rule-Based Detection (Decoders and Rules), Real-Time Alerting (Active Responses, Notifications).

Unit 5	**Secure Network Design and Implementation**	**9 hours**

Network Architecture Design Principles (segmentation, least privilege), Defense-in-Depth Strategy (multiple security layers, redundancy), Secure Routing Protocols (BGP security, OSPF authentication), Security Best Practices for LANs, WANs, and Wireless Networks (encryption, access control), Security Assessment and Auditing (vulnerability scanning, log analysis)

	Total Lecture Hours	**45 hours**

Textbook:

1. Network Security with OpenSSL: Cryptography by John Viega, Matt Messier, Pravir Chandra, published by O'Reilly Media **2.** Cryptography and Network Security: Principles and Practice by William Stallings, 7th Edition publishe Pearson Education		
Reference Books:		
1. Cryptography: Theory and Practice by Douglas Stinson, 4th Edition, CRC Press, taylor and Francis		
Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. Python Case Study - Cryptography - TOC | Infosys Springboard	5 Hrs 47 Min
Video Lectures	1. https://www.youtube.com/watch?v=Q-HugPvA7GQ&list=PL71FE85723FD414D7 https://onlinecourses.nptel.ac.in/noc22_cs90/preview	59 Min 2 Sec 21 Hrs
Web References	1. NIST Cryptographic Standards 2. NIST Computer Security Resource Center (CSRC) 3. RFC Editor Repository 4. OpenSSL Documentation 5. Wireshark Documentation 6. Cisco Networking Academy	2 Hrs 3 Hrs 15 min 1 Hrs 4 Hrs 5 Hrs 3 Hrs

Course Code: BCSAI0517	Course Name: Programming for Data Analytics	L	T	P	C
Course Offered in: **CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT**		3	0	0	3

Pre-requisite: Basic Knowledge of Python and R

Course Objectives:

Demonstrate knowledge of statistical data analysis techniques, apply Data Science principles, utilize data mining software, and employ modern Big Data tools and technologies to solve real-world business problems and support effective decision-making.

	Course Outcome: After completion of the course, the student will be able to	Bloom's Knowledge Level (KL)
CO1	Implement Python & R Programming Language and explain the use of R Studio IDE to perform basic tasks on Vectors, Matrices, and Data Frames.	K2
CO2	Apply the concepts of R packages to develop and manage data analysis applications.	K3
CO3	Apply the basic concepts of MongoDB to perform database operations in data-driven applications.	K3
CO4	Analyze the concepts of Recurrent Neural Networks (RNN) and TensorFlow for solving AI and deep learning problems.	K4
CO5	Analyze the role of Keras in developing and evaluating deep learning models.	K4

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

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	3	-	-	-	-	-	2	3	2	-
CO2	2	3	-	2	3	-	-	-	-	-	2	2	3	-
CO3	3	3	2	2	3	-	-	-	-	-	2	3	3	2
CO4	3	3	3	3	3	-	-	-	-	-	2	3	3	3
CO5	3	3	3	3	3	-	-	-	-	-	2	3	3	3

Course Contents / Syllabus

Unit- 1	BASIC DATA ANALYSIS USING PYTHON/R	09 hours

Pandas data structures — Series and Data Frame, Data wrangling using pandas, Statistics with Pandas, Mathematical Computing Using NumPy, Data visualization with Python, Descriptive and Inferential Statistics, Introduction to Model Building, Probability and Hypothesis Testing, Sensitivity Analysis, Regular expression: RE packages.

Unit- 2	R GRAPHICAL USER INTERFACES	10 hours

Built-in functions, Data Objects-Data Types & Data Structure, Structure of Data Items, Manipulating and Processing Data in R using Dplyr package &Stringr package, Building R Packages, Running and Manipulating Packages, data import and export, attribute and data types, descriptive statistics, exploratory data analysis, Flexdashboard and R-shiny.

Unit- 3	DATA ENGINEERING FOUNDATION	08 hours

Connecting to a database (sqlite) using Python, Sending DML and DDL queries and processing the result from a Python Program, Handling error, NOSQL query using MongoDB, MongoDB Compass.

Unit- 4	INTRODUCTION TO TENSORFLOW AND AI	09 hours
Introduction, Using TensorFlow for AI Systems, Up and Running with TensorFlow, Understanding TensorFlow Basics, Convolutional Neural Networks, Working with Text and Sequences, and TensorBoard Visualization, Word Vectors, Advanced RNN, and Embedding Visualization, TensorFlow Abstractions and Simplifications, Queues, Threads, and Reading Data, Distributed TensorFlow, Exporting and Serving Models with TensorFlow.		
Unit- 5	DEEP LEARNING WITH KERAS	09 hours
Introducing Advanced Deep Learning with Keras, Deep Neural Networks, Autoencoders, Generative Adversarial Networks (GANs), Improved GANs, Disentangled Representation GANs, Cross-Domain GANs, VariationalAutoencoders (VAEs), Deep Reinforcement Learning, Policy Gradient Methods.		
	Total Lecture Hours	**45 hours**

Textbook:

Book Details	Authors Name
1.	Glenn J. Myatt, Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining, John Wiley Publishers, 2007.
2.	Tom Hope, Yehezkel S. Resheff, Itay Lieder, Learning TensorFlow, O'Reilly Media, Inc.
3.	Advanced Deep Learning with TensorFlow 2 and Keras: Apply DL, GANs, VAEs, Deep RL, Unsupervised Learning, Object Detection and Segmentation, and More, 2nd Edition.

Reference Books:

Book Details	Authors Name
1.	Boris Lublinsky, Kevin T. Smith, Alexey Yakubovich, "Professional Hadoop Solutions", 1st Edition, Wrox, 2013.
2.	Chris Eaton, Dirk Deroos et al., "Understanding Big Data", Indian Edition, McGraw Hill, 2015.
3.	Tom White, "HADOOP: The Definitive Guide", 3rd Edition, O'Reilly, 2012.

Self-Study Content Links:		Duration (in Hrs)
MOOCs / Video Lectures / Web References		
Unit 1	https://www.ibm.com/cloud/blog/python-vs-r	
Unit 2	https://www.youtube.com/watch?v=CSR5SdYzOBI	
Unit 3	https://hevodata.com/learn/data-engineering-and-data-engineers/	
Unit 4	https://www.youtube.com/watch?v=lEZmH7byZO	
Unit 5	https://www.youtube.com/watch?v=pWp3PhYI-OU	01:11 Min

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

Course Code: BCSE0514	**Course Name: Design Patterns**	**L**	**T**	**P**	**C**
Course Offered in: CSE/IT/ CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(CYS)/CSE(IOT)/MCT		**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

CO	Course Outcome	Bloom's Level
CO1	Analyze, classify, and apply appropriate design patterns to solve real-world software problems, understanding their structure and selection principles.	K3
CO2	Identify, compare, and apply creational design patterns (Factory, Abstract Factory, Builder, Prototype, Singleton) to develop flexible and reusable software components.	K4
CO3	Analyze, design, and implement structural design patterns to improve flexibility, reusability, and maintainability of software systems.	K4
CO4	Analyze, design, and apply behavioral design patterns (Mediator, Memento, Observer) to manage object interactions and system behavior effectively.	K3
CO5	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	PSO2	PSO3
CO1	3	3	2	2	2	-	-	1	1	-	2	2	–	3
CO2	3	2	3	1	3	-	-	1	1	-	2	2	1	3
CO3	3	3	3	2	3	-	-	1	1	-	2	2	1	3
CO4	3	3	3	2	2	-	-	1	1	-	2	2	–	3
CO5	3	2	3	2	2	-	-	1	2	-	3	2	–	3

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	**Behavioural Patterns - I**	**9 hours**

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

Unit 5	**Behavioural Patterns - II**	**9 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:

S.NO.	**TITLE**	**Authors**
1	First Design Patterns, 2004, O'Reilly	Eric Freeman, Elisabeth Freeman, Kathy Sierra, Bert Bates Head
2	Design Patterns: Elements of Reusable Object oriented Software Addison-Wesley, 1995.	Erich Gamma, Richard Helm,

		Ralph Johnson, John Vlissides
Reference Books:		
S.NO.	**TITLE**	**Authors**
1	Design Patterns Pearson Education	Erich Gamma
2	Patterns in JAVA Volume -I	Mark Grand, Wiley Dream
Self Study Content Link:		
MOOCs	1. https://www.coursera.org/learn/design-patterns 2. https://www.udemy.com/course/soft-design-patterns/?utm_campaign=Search_DSA_Beta_Prof_la.EN_cc.India&utm_source=google&utm_medium=paid-search&portfolio=India&utm_audience=mx&utm_tactic=nb&utm_term=_._ag_160270535505_._ad_696202838334_._kw_&utm_content=g&funnel=&test=&gad_source=1&gad_campaignid=21178559974&gbraid=0AAAADROdO3wWjcAcJrZ3ow_Fcj2OFn24&gclid=Cj0KCQjw3K7RBhDJARIsAKRtP5TgfS7r7XJhzwC-gZgMsTFRkUx44pchkPY7WDXEXQyxDGeidGMqrbAaAn56EALw_wcB&couponCode=PMNVD2025	25 hrs 5 hrs
Video Lectures	1. https://www.youtube.com/watch?v=ysoC8S96pZU&list=PLBBog2r6uMCResrR48HziFtNx4-BDOmU0 2. https://www.youtube.com/playlist?list=PLmCsXDGbJHdhULIoYzIIIeFFJSo1NlMbN 3. https://www.youtube.com/watch?v=dCWuXDBRFtk&list=PL2NZAYdLkYvhFnqbvLbwf4ZM4rH99Jap6 4. https://www.youtube.com/watch?v=a4u_caEWTR8&list=PL2NZAYdLkYvg__5SpIDMlRopz-dD_KsIX	36 Min 1 hr 6 Min 8 Min 50 Sec
Web References	1. https://refactoring.guru/design-patterns 2. https://www.geeksforgeeks.org/system-design	

Course Code: BCSAI0519N	Course Name: Business Intelligence and Data Visualization	L	T	P	C
Course Offered in: CS/CSE(IOT)		3	0	0	3

Prerequisite: Basic Knowledge of Excel, Statistics and Computer Skills.

Course Objectives:

Present the concepts, architecture, and lifecycle of Business Intelligence systems, develop skills in data preparation and modeling using BI tools, create interactive visualizations and dashboards, apply analytical techniques for data-driven decision-making, and build end-to-end Business Intelligence solutions using real-world case studies and projects.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the fundamental concepts, architecture, and components of Business Intelligence, Query, Reporting, and Development systems.	K2
CO2	Apply data modeling, ETL/ELT processes, and data preparation techniques using BI tools like Tableau and Power BI.	K3
CO3	Analyze datasets using visualization techniques, OLAP operations, and interactive reporting features such as drill-down and drill-up.	K4
CO4	Evaluate business problems using advanced analytics, dashboards, KPI metrics, and decision-support systems for meaningful insights.	K5
CO5	Design end-to-end BI solutions including dashboards, reports, and mini-projects using real-world datasets and modern BI platforms.	K6

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

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	1	1	1	1	-	1	1	1	1	2	1	1
CO2	2	2	2	2	3	1	-	1	1	1	1	2	3	1
CO3	1	3	2	3	3	1	-	1	1	2	1	3	3	1
CO4	1	3	2	3	3	2	-	2	1	2	2	3	3	2
CO5	2	2	3	3	3	2	-	2	3	3	3	3	3	3

Course Contents / Syllabus

Unit- 1	Business Intelligence, Query, Reporting and Development	10 hours

Introduction to Business Intelligence (BI) and Business Analytics; Evolution, scope, architecture, and components of BI systems; Data sources, data warehousing, and ETL/ELT processes; Cloud data warehousing, lakehouse architecture, OLAP, and multidimensional modeling; Advanced analytics, predictive analytics, AI/ML integration, real-time and self-service BI; Data visualization, dashboards, reports, scorecards, and interactive BI development including drill-down and drill-up techniques; Functional applications of BI tools and platforms such as Tableau, Power BI, Qlik, and Looker; KPI monitoring, decision support systems, and performance measurement; Big data analytics, Hadoop ecosystem, IoT, and generative AI applications in BI; Data governance, metadata management, security, and privacy; BI project lifecycle including design, development, deployment, reporting, automation, mobile BI, SDK usage, risk management, and success evaluation; BI adoption challenges, industry trends, and real-world Industry 4.0 applications.

Unit- 2	Introduction to TABLEAU	8 hours

Overview of Tableau ecosystem with focus on Tableau Public; Installation, interface, and workspace navigation; Connecting to data sources (Excel, CSV, Google Sheets, web data); Data preparation basics (cleaning, filtering, joins, unions); Understanding data types and dimensions/measures; Basic charts (bar, line, pie, scatter plots); Sorting, filtering, grouping, and aggregation; Introduction to calculated fields and simple expressions; Building basic dashboards in Tableau Public; Publishing and sharing visualizations online; Case study: Sales performance dashboard using public dataset; Lab work: Creating basic charts and publishing to Tableau Public profile.

Unit- 3	Advanced Visualization, Dashboards and Projects in Tableau	9 hours
Advanced chart types (maps, heat maps, tree maps, dual-axis charts); Interactive dashboards and dashboard actions (filter, highlight, URL actions); Parameters and advanced calculated fields; Data blending and multiple data sources; Storytelling with Tableau (story points and narratives); Introduction to Level of Detail (LOD) expressions; Performance optimization and best practices; Real-world BI dashboards for business domains (finance, marketing, healthcare, supply chain); Case studies: KPI dashboard for executive reporting and COVID/stock market visualization using public datasets; Lab work: End-to-end dashboard development and publishing on Tableau Public; Mini project: Design and deploy a complete interactive BI solution using real-world dataset with documentation and presentation.		
Unit- 4	**Introduction to POWER BI and Data Preparation**	**8 hours**
Introduction to Business Intelligence and Power BI ecosystem; Overview of Power BI Desktop, Service, and Mobile; Installation and interface navigation; Connecting to data sources (Excel, CSV, databases, web data); Data loading and transformation using Power Query Editor; Data cleaning, shaping, merging, and appending datasets; Data modeling concepts (relationships, star schema basics); Introduction to DAX (Data Analysis Expressions); Basic measures and calculated columns; Creating simple visuals (bar, line, pie, table, map); Filters, slicers, and report interactions; Publishing reports to Power BI Service; Case study: Sales data analysis dashboard; Lab work: Data cleaning and basic report creation using real dataset.		
Unit- 5	**Advanced Power BI, Dashboards and Project Development**	**10 hours**
Advanced DAX functions (CALCULATE, FILTER, time intelligence); Advanced visualizations (KPI cards, drill-through, decomposition tree, funnel charts); Interactive dashboards and report design best practices; Data modeling optimization and relationships management; Power BI Service features (workspaces, sharing, collaboration, refresh scheduling); Row-level security and data governance basics; Performance optimization techniques; Real-world business dashboards (finance, HR, marketing, operations); Case studies: Executive KPI dashboard and retail performance analysis; Lab work: End-to-end dashboard development and deployment; Mini project: Design a complete Power BI solution using real-world dataset with dashboard publishing, documentation, and presentation.		
	Total Lecture Hours	**45 hours**

Textbook:

1. "Business Intelligence Guidebook: From Data Integration to Analytics" – Rick Sherman
2. "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling" – Ralph Kimball & Margy Ross.
3. "Business Intelligence in Plain Language: A Practical Guide to Data Mining and Business Analytics" – Jeremy Kolb

Reference Books:

1. Learning Tableau 2025 (6th Edition) – Joshua N. Milligan.
2. The Definitive Guide to DAX – Marco Russo & Alberto Ferrari

NPTEL/ YouTube/ Faculty Video Link

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. https://www.tableau.com/learn/training 2. https://www.coursera.org/specializations/data-visualization	60 Hrs.
Video Lectures	1. https://www.youtube.com/watch?v=aHaOIvR00So 2. https://www.youtube.com/watch?v=4RW7JgbQ4ao	12 Hrs.
Web References	1. https://learn.microsoft.com/en-us/power-bi/ 2. https://trailhead.salesforce.com/trailblazer 3. community/neighborhoods/tableau	15 Hrs.

Course Code: BCSCY0515	Course Name: Ethical Hacking	L	T	P	C
Course Offered in: CSE/IT/CSE-r/ M.Tech./ M.Tech.(INT.)/ CSE(AI)/ CSE(AIML)/ CSE(DS)/ CS/ CSE(IOT) / MCT		3	0	0	3

Prerequisite: Basic understanding of computer networks, operating systems, cybersecurity fundamentals, and programming concepts.

Course Objectives:

Equip participants with hands-on skills in ethical hacking methodologies from reconnaissance to reporting enabling them to identify, exploit, and remediate vulnerabilities in real-world environments.

Course Outcome: After completion of this course students will be able to		Bloom's Knowledge Level (KL)
CO1	Explain ethical hacking concepts and apply basic ethical hacking techniques in practical scenarios.	K2
CO2	Describe and use tools for footprinting, reconnaissance, and port scanning activities.	K2, K3
CO3	Identify and examine security vulnerabilities in web applications using ethical hacking tools and techniques.	K2, K3
CO4	Explain scanning and enumeration concepts for identifying network resources and vulnerabilities using OpenVAS.	K3
CO5	Describe system hacking techniques such as unauthorized access, privilege escalation, and persistent control along with defense methods.	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	1	-	1	-	1	-	2	-	2	1
CO2	3	3	2	2	2	-	2	-	2	-	2	2	3	2
CO3	3	3	3	2	3	-	3	2	2	1	3	3	3	3
CO4	2	2	2	1	2	1	3	2	2	-	3	-	3	3
CO5	2	2	2	2	1	1	3	2	2	1	3	-	2	3

Course Contents / Syllabus

Unit- 1	Introduction to Ethical Hacking.	9 hours

Introduces the origins and progression of hacking, ethical guidelines and code-of-conduct, applicable legal frameworks (IT Act, GDPR), core terminologies (white-hat, black-hat, APT), and provides hands-on orientation to essential toolkits like Kali Linux and Metasploit.

Unit- 2	Foot printing & Reconnaissance	8 hours

OSINT techniques; Passive/active reconnaissance; DNS interrogation (NS lookup, dig command); WHOIS lookup; Social engineering; Vulnerability scanning.

Unit- 3	Web Application Security	10 hours

Introduction to Web Application Security, Common Web Vulnerabilities, Owasp: Broken Access Control, Securities Misconfiguration, Software Supply Chain failures, Cryptographic failures, Injection, Insecure Design, Authentication failures, Software data integrity failures, Securities logging and alerting failures, Mishandling of exceptional conditions, Web Application Reconnaissance and Foot printing, Web Application Scanning and Enumeration, Secure Coding Practices and Web Application Hardening.

Unit- 4	Network Attacks and Défense Mechanisms	8 hours

Port Scanning Techniques (TCP, UDP), Nmap Tool, Service Enumeration and Version Detection, OS Fingerprinting, Vulnerability Assessment and Analysis Using Scanning Tools like, Nessus, and OpenVAS..

Unit- 5	Defensive strategies	10 hours
Introduction to Password Cracking Tools & Techniques, Wi-Fi Password Cracking Techniques, Escalating Privileges and Gaining Unauthorized Access, Exploiting System Vulnerabilities, Introduction to Malware Analysis & Reverse Engineering, Countermeasures and Defensive Strategies against System Hacking.		
	Total Lecture Hours	**45 hours**

Textbook:

1. The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy By Patrick Engebretson. 2nd Edition Elsevier / Syngress
2. CEH Certified Ethical Hacker All-in-One Exam Guide By Matt Walker. 2th Edition, McGraw Hill.
3. Web Application Hacker's Handbook: Finding and Exploiting Security Flaws By Dafydd Stuttard and Marcus Pinto. 2nd Edition, Wiley Publishing

Reference Books:

1. Penetration Testing: A Hands-On Introduction to Hacking, Georgia Weidman. 1st Edition, No Starch Press

NPTEL/ YouTube/ Faculty Video Link:

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

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

Pre-requisite: Fundamental of Computers.

Course Objectives:

Understand and configure fundamental networking concepts, develop practical skills in IP addressing, subnetting, VLANs, routing, switching, and wireless networking, configure and manage essential network services, perform network testing and troubleshooting using Cisco IOS tools, and gain hands-on experience aligned with industry networking standards and Cisco CCNA certification.

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

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 addresses and ARP work in local communication.	CO1
10. Basic Switch Configuration Configure switch settings, ports, and management IP.	CO2
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: 60 hrs.	

LAB Course Code: BCSML551B	LAB Course Name: Machine Learning Algorithms Lab	L	T	P	C
Course Offered in: B. Tech. 3rd year (5th Semester) Computer Science		0	0	2	1

Pre-requisite: Basic knowledge of Python programming and Statistics fundamentals

Course Objectives:

Develop practical skills in designing, training, evaluating, and deploying deep learning and machine learning models, perform data preprocessing and feature engineering, implement supervised and unsupervised learning techniques, evaluate model performance using appropriate validation methods, and apply modern AI frameworks to solve real-world problems.

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

CO1	Apply basic data exploration and preprocessing techniques to identify dataset characteristics, missing values, duplicate records, and outliers.	K3
CO2	Analyze the dataset to determine appropriate strategies for handling missing values, categorical data imputation, and data inconsistencies.	K4
CO3	Evaluate and implement suitable data cleaning methods, including row removal, value imputation, duplicate elimination, and outlier treatment, to prepare a high-quality dataset for analysis.	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	2	2	3	1	-	3	1	-	2	3	2	3
CO2	3	2	3	3	3	1	-	3	1	1	2	3	2	3
CO3	3	3	3	3	3	1	-	3	1	1	3	3	3	3

List Of Practical's (Indicative & Not Limited To)	CO
1. Perform the following on a dataset: a) Basic data exploration, check for missing values, remove rows with missing values, Fill missing numerical values with mean or median, fill in categorical values with mode or a specific value, Handling duplicate rows, Removing outliers. b) Dealing with categorical Variables: one-hot encoding, label encoding, ordinal encoding, and target encoding c) Feature Scaling: MinMaxScaler, StandardScaler, MaxAbsScaler, RobustScaler, Normalizer d) Correlation Analysis 2. Implement Principal Component Analysis (PCA) algorithm for dimensionality reduction.	CO1
1. Implement a class having functions for Mean Absolute Error, Root Mean Square Error, Log loss, R-square and Adjusted R Square 2. Fit a linear regression model to predict housing prices based on the size of the house. 3. Forecasting Stock Closing Prices with Polynomial Regression and Bias-Variance Trade-off Analysis. 4. Implement Gradient Descent algorithm and analyse the effect of learning rate and derivatives. 5. Demonstrate effect of underfitting and overfitting in already implemented regression algorithms.	CO2
1. Implement logistic regression for binary classification on a dataset. 2. Implement logistic regression model for Handwritten digit classification. 3. Implement K-Nearest Neighbor regression from scratch for classification. 4. Use the ID3 algorithm to build a decision tree to predict whether a customer will purchase a product based on their browsing behaviour on an e-commerce website. 5. Use a support vector machine (SVM) to classify images into different categories using the CIFAR-10 dataset.	CO2
1. Implement Email Spam Classifier using Naïve Bayes Algorithm. 2. Implement Random Forest algorithm and Xgboost algorithm for Telco Customer Churn Prediction and compare the performance of both the algorithms.	CO3
1. Customer Segmentation for a Retail Store Using K-Means Clustering. 2. Document Clustering Using Hierarchical Clustering and Dendrograms. 3. Fraud Pattern Detection in Transactions Using DBSCAN Clustering. 4. Clustering City Temperature Patterns Using Gaussian Mixture Models. 3. Implementation of anomaly detection system.	CO3
	Total Hours: 30Hrs

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

1. Students will be able to develop dynamic web applications using JDBC, Servlets, and JSP with proper database connectivity and session management.
2. Students will apply Spring Framework concepts, including IOC and Dependency Injection, to build structured and scalable applications.
3. 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	-
CO2	3	3	3	2	2	1	-	3	1	1	2	3	3	-
CO3	3	3	3	2	2	1	-	3	1	1	2	3	3	3

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 Application Context 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 Unit- s 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.
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

LAB Course Code: BCSCY0513P	Course Name: Network Security and Cryptography Lab	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		0	0	2	1

Pre-requisite: Understanding basic Networking Terms (e.g., IP, LAN, Router), Awareness of Digital Tools (Email, Cloud, Antivirus), Fundamental Logical Thinking, Cybersecurity fundamentals.

Course Objectives:

This course aims to introduce the fundamental principles and concepts of network security and cryptography, provide insights into common network-based attacks and vulnerabilities and develop the ability to design secure network architectures using modern cryptographic tools and techniques. Provide hands-on experience with classical and modern cryptographic algorithms.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Demonstrate the implementation and analysis of classical and modern cryptographic techniques including shift cipher, monoalphabetic cipher, one-time pad, MACs, hash functions, AES, DES, RSA, and ECC for secure data communication.	K3
CO2	Configure and evaluate secure communication protocols and network security mechanisms such as SSL/TLS, TLS handshake analysis, IPsec VPNs, and firewall configurations using appropriate security tools and techniques.	K4
CO3	Analyze and perform network security assessment activities including Man-in-the-Middle (MITM) attack simulation, DoS attack simulation, and penetration testing to identify vulnerabilities and strengthen system security.	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	-	-	-	2	3	1
CO2	3	3	2	2	3	1	-	1	1	1	1	3	2	2
CO3	3	3	1	2	2	-	-	1	-	-	-	1	3	3

List Of Practical's (Indicative & Not Limited To)	CO Level
1. Implementing Shift Cipher (Caesar Cipher).	CO1
2. Implementing Monoalphabetic Substitution Cipher.	CO1
3. Implementing Vigenère Cipher.	CO1
4. Implementing One-Time Pad and Perfect Secrecy.	CO1
5. Implementing Message Authentication Codes (MAC).	CO1
6. Implementing Cryptographic Hash Functions (SHA-256).	CO1
7. Implementation of AES Encryption Algorithm.	CO1
8. Implementation of DES Encryption Algorithm using Python.	CO1
9. Implementation of RSA Algorithm (Key Generation, Encryption/Decryption).	CO1
10. Digital Signature Generation and Verification using RSA/ECC.	CO1
11. Packet Capturing and Analysis using Wireshark.	CO2
12. Basic Firewall Configuration using IPTables/pfSense.	CO2
13. Vulnerability Scanning using Nmap/OpenVAS.	CO2
14. Risk Assessment and Security Policy Documentation for a Small Network.	CO2
15. Configuration of SSL/TLS for Secure Web Communication.	CO2
16. TLS Handshake Analysis using Wireshark.	CO2
17. Secure Remote Login using SSH and Key-Based Authentication.	CO2
18. SSH Port Forwarding and Secure Tunneling.	CO2
19. Setting up IPsec VPN for Secure Network Communication.	CO2

20. Wireless Security Configuration using WPA2/WPA3.	CO2
21. DoS Attack Simulation in a Safe Lab Environment.	CO3
22. Man-in-the-Middle (MITM) Attack Simulation.	CO3
23. Intrusion Detection using Snort IDS.	CO3
24. OSSEC Installation, Log Monitoring, and Alert Configuration.	CO3
25. Secure Network Design using VLAN Segmentation and Access Control.	CO3
26. Network Security Assessment and Penetration Testing.	CO3
Total Hours: 30 hrs.	

Course Code: BCSAI0517P	Course Name: Programming for Data Analytics Lab	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		0	0	2	1

Pre-requisite: Basic Knowledge of Python and R

Course Objectives:

Develop practical skills in data analysis, wrangling, visualization, statistical analysis, and regression using Python and R, build interactive dashboards and web applications, perform database operations using SQLite and MongoDB, and apply basic AI techniques using TensorFlow to solve real-world data-driven problems.

Course Outcome:	After completion of the course, the student will be able to	Bloom's Knowledge Level (KL)
CO-1	Apply Python libraries (Pandas, NumPy, Matplotlib, Scikit-learn) to perform data wrangling, statistical computation, visualization, and basic regression modeling.	K3
CO-2	Implement statistical hypothesis testing and R programming constructs, including packages, EDA, dashboards, R-Shiny apps, and SQLite database operations.	K3
CO-3	Implement NoSQL database operations using MongoDB and build basic AI systems using TensorFlow.	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	-	2	3	-	-	-	-	-	1	3	2	-
CO2	2	3	1	2	3	-	-	-	-	-	1	2	3	-
CO3	3	3	2	2	3	-	-	-	-	-	1	3	3	2

List of Programs

S.No	List of Program	CO
1	Create a Pandas Series and DataFrame, perform indexing, slicing, filtering, and handle missing values.	CO1
2	Perform data wrangling using Pandas – merge, reshape, group, and aggregate data from a dataset.	CO1
3	Perform mathematical computing using NumPy – array operations, matrix multiplication, and statistical functions.	CO1
4	Visualize data using Matplotlib – draw line plot, bar chart, histogram, scatter plot, and pie chart.	CO1
5	Perform descriptive statistics (mean, median, mode, std, variance) and inferential statistics on a dataset.	CO1
6	Fit a simple linear regression model using Scikit-learn and visualize the regression line.	CO1
7	Perform hypothesis testing (t-test) and calculate confidence intervals for a sample dataset.	CO2
8	Perform a chi-squared test for independence on categorical data and interpret the p-value.	CO2
9	Perform sensitivity analysis on a function by varying input parameters and observing output changes.	CO2
10	Extract phone numbers and email addresses from a text string using Regular Expression (RE) packages.	CO2
11	Demonstrate R built-in functions: mean(), sum(), median(), length(), sd() on a sample dataset.	CO2
12	Create and manipulate R data structures – vectors, lists, matrices, arrays, and data frames.	CO2
13	Explore structure of data items in R using str(), typeof(), class(), and mode() functions.	CO2

14	Use the dplyr package to filter, arrange, select, mutate, and summarize data from a dataset.	CO2
15	Use the stringr package for pattern matching, string replacement, substring extraction, and case transformation.	CO2
16	Install, load, update, and uninstall R packages (ggplot2, dplyr, tidyr) and demonstrate their usage.	CO2
17	Read data from a CSV file into R, manipulate using dplyr, and export the processed data back to CSV.	CO2
18	Perform Exploratory Data Analysis (EDA) using summary statistics, visualizations, and correlation checks in R.	CO2
19	Create an interactive dashboard using the flexdashboard package to visualize key statistics of a dataset.	CO2
20	Develop a basic interactive web application using R Shiny with user inputs and dynamic plot/table outputs.	CO2
21	Write a Python program to connect to an SQLite database and create a table using DDL queries.	CO2
22	Write a Python program to send SELECT queries to an SQLite database and display the processed results.	CO2
23	Write a Python program to insert, update, and delete records in an SQLite table using DML queries.	CO2
24	Write a Python program to handle SQLite database connection and query errors using try-except blocks.	CO3
25	Write a Python program to connect to MongoDB using pymongo and retrieve a list of all collections.	CO3
26	Write a Python program to insert single and multiple documents into a MongoDB collection.	CO3
27	Write a Python program to query MongoDB data using find(), find_one(), and various filter conditions.	CO3
28	Write a Python program to update and delete MongoDB documents using update_one(), update_many(), delete_one(), and delete_many().	CO3
29	Write a Python program using MongoDB aggregation framework to perform grouping, counting, and summing operations.	CO3
30	Explore a MongoDB database using MongoDB Compass – view, filter, update, and delete data through the GUI.	CO3
31	Create a TensorFlow program to perform basic mathematical operations – tensor creation, matrix multiplication, and addition.	CO3
32	Build a basic AI system using TensorFlow that predicts simple outputs from input data (linear regression model).	CO3
Total Hours: 30 hrs.		

Course Code: BCS0501	**Course Name: Cloud Architecture and Infrastructure**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech. 3rd year (5th Semester) Computer Science		**0**	**0**	**6**	**3**

Pre-requisite: Fundamental of Cloud Computing, Computer Networks

Course Objectives:

Develop a comprehensive understanding of cloud computing concepts and architecture, design scalable, secure, and reliable clo ud-based solutions, deploy and manage cloud infrastructure using industry-standard tools, optimize cloud resources for performance and cost efficiency, and apply modern cloud technologies to real-world deployment scenarios.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO-1	Develop a comprehensive understanding of federal cloud infrastructure and its web services.	K2
CO-2	Analyze Cloud Computing Reference Architectures and challenges in open architecture.	K4
CO-3	Deploy Amazon EC2 applications, ensuring reliability and cost-effectiveness under varying workloads.	K3
CO-4	Deploy cloud network architecture solutions to ensure secure, scalable, and efficient connectivity.	K3
CO-5	Implement security mechanisms and identity management strategies within cloud computing environments to ensure data protection and controlled access.	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	-	-	2	–	–	-	-	-	–	3	1	1
CO2	3	3	2	2	2	–	–	-	–	-	–	3	2	1
CO3	3	3	3	2	3	-	1	-	1	-	-	3	3	2
CO4	3	3	3	2	3	-	2	-	2	1	–	3	3	2
CO5	3	3	2	2	3	2	2	3	1	-	2	3	3	2

Course Contents / Syllabus

Unit 1	**Introduction to Cloud Computing Architecture**	**14 hours**

Definition, provisioning and manageability of cloud computing, underlying principles of parallel and distributed computing. AWS global infrastructure (for example, Availability Zones, AWS Regions), AWS federated access and identity services, SOA, Web services, RESTful services, and Publish-subscribe model.

Unit 2	**Cloud computing reference architectures:**	**14 hours**

Introduction to Cloud Computing Reference Architecture (CCRA): CCRA Evolution, Benefits of CCRA:
Architectural overview: The conceptual Reference Model, Cloud Consumer, Cloud provider, Cloud Auditor, Cloud carrier, Scope of control between Provider and Consumer.
Functional and Non-functional requirements of cloud architecture: IBM's Cloud Computing Reference Architecture (CCRA 4.0), Introduction, Roles, Architectural Elements.

Unit 3	**Server and Storage architecture:**	**14 hours**

Selection of appropriate compute options, features, type and size for deployment of EC2, Scalability capabilities for Amazon EC2 Auto Scaling and AWS Auto Scaling, Serverless technologies and patterns with AWS Lambda and AWS Fargate, and Serverless Architecture.
AWS Storage Services: Elastic Block Storage (EBS), Elastic File Storage (EFS), Amazon S3, Amazon RDS, and DynamoDB to support scalable and reliable data management.

Unit 4	**Network Architecture**	**15 hours**

Networking Fundamentals: Subnets, Routing, Security Groups, DNS, Federated Access, ACL, NAT, VPC, VPC architecture, Network segmentation strategies, VPN, AWS Direct Connect, VPC Peering, VPC Endpoint, Security Groups, AWS Route 53

Unit 5	Security Architecture	15 hours
AWS Network Firewall, AWS Firewall Manager, Applying AWS security best practices to IAM users and root users (Ex: Multi-Factor Authentication), authorization model that includes IAM users, groups, roles, and policies, role-based access control strategy (Ex: AWS Security Token Service [AWS STS], role switching, cross-account access, AWS Key Management Service, AWS Resource Access Manager.		
	Total Lecture Hours	**72 hours**

Textbook:	
Books Details:	**Authors Name**
'Mastering Cloud Computing', Mc Graw Hill 2013	Raj Kumar Buyya
Solutions Architect's Handbook, Third Edition, Packt, 2024	Saurabh Shrivastava, Neelanjali Srivastav

Reference Books:	
Books Details:	**Authors Name**
'Cloud Computing' Oxford higher education 2022	Shailendra Singh
AWS for Solutions Architects, Packt Publishing, 2021	Alberto Artasanchez

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs		14Hrs 09Min
Video Lectures	1. https://nptel.ac.in/courses/106/105/106105167/ 2. https://nptel.ac.in/courses/106/105/106105223/ 3. https://aws.amazon.com/docs 4. https://nptel.ac.in/courses/106/104/106104182 5. https://www.nist.gov/	9Hrs 36Min 6Hrs 36Min 23Hrs 26Min 25Hrs 05Hrs 35Min
Web References	https://docs.aws.amazon.com/ https://aws.amazon.com/training/ https://cloud.google.com/docs https://learn.microsoft.com/azure/	

List Of Practical's (Indicative & Not Limited To)
1. Launch a Linux EC2 instance and demonstrate provisioning and basic manageability.
2. Create static website and deploy it on S3.
3. Boot EC2 windows instance using Remote Desktop Protocol into DSRM.
4. Design and deploy a simple web service on Amazon EC2.
5. Launch a MySQL RDS instance.
6. Create a VPC with a valid CIDR block.
7. Create a public and private subnet in the VPC
8. Configure and build a RDS server inside a private cloud.
9. Configure and establish an internet connection to the instance in the private cloud using a NAT gateway.
10. Configure the security group and access control with an inbound and outbound rule for an instance available in a public subnet.
11. Configure the security group and access control with an inbound and outbound rule for an instance available in the private subnet.
12. Implement Auto Scaling with a Load Balancer and CloudWatch alarms for an EC2-based web app.

Course Code: BCSCC0501Y	Course Name: Design Thinking –II	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(CYS)/CSE(IOT)/MCT		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 idea 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)

Course Contents/ Syllabus

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	2	3	3	2	2	2	3	2	3	3	-	-
CO2	3	2	3	2	3	2	2	3	2	2	2	3	3	-
CO3	3	2	3	2	3	2	2	3	2	2	2	3	3	-
C04	3	2	3	2	3	2	2	3	2	2	2	3	3	-
CO5	3	2	3	3	3	2	2	3	3	2	3	3	3	-

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

Self-Study Content Links:

MOOCs	1. https://www.coursera.org/specializations/design-thinking?utm_medium=sem&utm_source=gg&utm_campaign=b2c_india_x_multi_ftco f_career-academy_cx_dr_bau_gg_pmax_gc_in_all_m_hyb_24-03_desktop&campaignid=21104989118&adgroupid=&device=c&keyword=&matchtype=&network=x&devicemodel=&creativeid=&assetgroupid=6544944776&targetid=&extensionid=&placement=&gad_source=1&gad_campaignid=21104990591&gbraid=0AAAADdKX6YJHt3GyzupmRYbPOgHkEeFp&gclid=Cj0KCQjw3K7RBhDJARIsAKRtP5RBjckHBvGEYiEnTRFA8EtXpl89ha-X8id2TOYd0e4Sf9WTvVnonK4aAmqrEALw_wcB 2. https://www.coursera.org/learn/uva-darden-designbiz	4 weeks 6 hrs
Video Lectures	1. https://www.youtube.com/watch?v=6_mHCOAAEI8 https://nptel.ac.in/courses/110106124 https://designthinking.ideo.com/ 2. https://www.youtube.com/watch?v=7enWesSofhg 3. https://youtu.be/rthuFS5LSOo https://www.youtube.com/watch?v=kho6oANGu_A 4. https://www.youtube.com/watch?v=9vMpHk44XXo&list=PL1xHD4vteKYVpaliy295 pg6_SY5qznc77&i 5. Reinforcement Learning Tutorial | Reinforcement Learning Example Using Python| Edureka - YouTube 6. Association Rule Mining – Solved Numerical Question on Apriori 7. Algorithm(Hindi) - YouTube Q Learning Explained | Reinforcement Learning Using Python | Q Learning in AI | Edureka - YouTube	7 hrs
Web References	https://blog.experiencepoint.com/how-mcdonalds-evolved-with-design-thinking	

Course Code: BCSCC0552	Course Name: Research Methodology Ethics and Writing	L	T	P	C
Course Offered in: All branches		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		Bloom's Knowledge Level (KL)
CO1	Analyze research methodologies, experimental design principles, and literature review techniques for identifying research gaps.	K3
CO2	Apply quantitative and qualitative data acquisition, analytical methods, technical documentation, and scientific citation practices in research activities.	K4
CO3	Develop technical reports, research manuscripts, and professional presentations using standard scientific writing and ethical communication practices.	K2

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	–	1	–	2
CO2	2	3	1	3	2	–	–	2	1	2	–	2	1	2
CO3	2	3	2	3	2	–	–	3	2	3	1	1	–	1

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:

SNO	Book Title	Author
1.	Research Methodology: Methods and Techniques	C.R. Kothari & Gaurav Garg
2.	Research Methodology: A Step-by-Step Guide for Beginners	Ranjit Kumar

3.	The Craft of Scientific Writing	Michael Alley
4.	How to Write and Publish a Scientific Paper	Day & Gastel
5.	Research Methodology and Scientific Writing	George Thomas

Reference Materials	
S.NO	
1.	IEEE Author Guidelines
2.	Elsevier Researcher Academy Resources
3.	Springer Author Academy
4.	WIPO Resources for Research Documentation
5.	University digital library resources

Self Study Content Links:

MOOCs Courses	1. https://www.coursera.org/learn/research-methods 2. https://researcheracademy.elsevier.com/	6 hrs 20 hrs
Video Lectures	1. https://www.youtube.com/watch?v=rz30rRfManE&list=PLdj5pVg1kHiOypKNUmO0NKOfvoIThAv4N 2. https://www.youtube.com/watch?v=HYj4Ght1_qs 3. https://www.youtube.com/watch?v=N9tJCs-czOU&list=PLnIj1GYixjfrOhcWQq_ANzLFDniGYZ_-Q 4. https://www.youtube.com/watch?v=NM53k7x_jjk	17 hrs
Web References	https://researcheracademy.elsevier.com https://authorservices.wiley.com	

SYLLABUS- SEMESTER VI

Course Code: BCSML0601B	**Course Name: Deep Learning Techniques**	**L**	**T**	**P**	**C**
Course Offered in: B. Tech. 3rd Year (6th Semester) Computer Science (CS)		**3**	**1**	**0**	**4**

Pre-requisite: Machine Learning, Linear Algebra, Probability and Statistics

Course Objectives:

Develop the ability to design, train, evaluate, and deploy deep learning models, apply ANN, CNN, and RNN architectures, preprocess data for image, speech, and text applications, optimize model performance using advanced training techniques, and implement intelligent solutions using modern deep learning frameworks and cloud-based AI platforms.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Explain the fundamentals of deep learning architectures, optimization methods, and regularization techniques for intelligent systems.	K2
CO2	Apply convolutional neural network models for image classification and computer vision applications.	K3
CO3	Implement object detection, tracking, and sequence modeling techniques for real-world AI applications.	K4
CO4	Analyze deep learning models for natural language processing, multilingual AI, and sequential data processing.	K4
CO5	Examine transformer architectures, autoencoders, and generative AI models for scalable cloud-based intelligent 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	2	1	3	–	–	1	1	–	2	3	2	1
CO2	3	3	3	1	3	–	–	1	1	–	2	3	3	1
CO3	3	3	3	3	2	1	–	2	1	1	2	3	3	2
CO4	3	3	2	3	2	1	–	1	2	–	3	3	3	1
CO5	3	3	3	3	3	1	1	1	2	1	3	3	3	1

Course Contents / Syllabus

Unit-1 **Deep Learning Foundations and Cloud-based Optimization** **10 hours**

Introduction to Deep Learning: Limitations of traditional Machine Learning, evolution from shallow learning to deep learning, Artificial Neural Networks (ANN), Feed Forward Neural Networks, multilayer perceptron architecture, forward propagation, activation functions (Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax), loss functions, backpropagation algorithm, optimization techniques: Stochastic Gradient Descent (SGD), Mini-batch Gradient Descent, Momentum, RMSProp, Adam optimizer, regularization techniques for deep learning: Dropout, Batch Normalization, Early Stopping, Data Augmentation, introduction to cloud-based model training, GPU/TPU accelerated computing environments for scalable deep learning.

Unit-2 **Convolutional Neural Networks and Cloud Computer Vision** **10 hours**

Introduction to Computer Vision and Convolutional Neural Networks (CNN): Convolution operation, kernels and feature maps, padding and stride, pooling operations (Max Pooling and Average Pooling), CNN architecture design, training convolutional neural networks, hyperparameter tuning in CNN, visualization and interpretation of CNN models, transfer learning and fine-tuning, popular **CNN architectures:** LeNet, AlexNet, VGG, ResNet, cloud-based image classification workflows, deployment considerations for computer vision models in scalable AI platforms.

Unit-3 **Object Detection, Recognition and Edge-Cloud AI Systems** **8 hours**

Object Detection Fundamentals: Bounding Box and Localization, Intersection over Union (IoU), evaluation metrics: Precision, Recall, F1-score, mean Average Precision (mAP), real-time object detection models: YOLO, SSD, advanced detection architectures: YOLOv8, RetinaNet, object tracking concepts, tracking-by-detection approach, SORT algorithm, Deep SORT algorithm, deployment considerations for real-time inference in edge and cloud AI environments, applications in surveillance, autonomous systems, and smart vision applications.

Unit-4	Sequence Modeling, Natural Language Processing and Cloud AI Services	8 hours
Sequential Data and Sequence Modeling: Recurrent Neural Networks (RNN), recurrent network architecture, backpropagation through time (BPTT), vanishing gradient problem, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Bidirectional RNN, Deep RNN, sequence-to-sequence models. **NLP:** Word embeddings and vector representations, text preprocessing for NLP, sentiment analysis, language modeling, sequence prediction and generation, multilingual NLP applications, introduction to cloud-based NLP services and AI-driven language applications		
Unit-5	**Transformers, Autoencoders and Generative AI Applications**	**9 hours**
Autoencoders in Deep Learning: definition, motivation, architecture, loss functions, sparse autoencoder, denoising autoencoder, stacked autoencoder, variational autoencoder (VAE), sequence autoencoder, feature representation and dimensionality reduction using autoencoders. **Transformer Architecture:** attention mechanism, self-attention, encoder-decoder architecture. **Large language models (LLMs):** foundation models, prompt engineering basics, generative AI applications, cloud deployment concepts for foundation models, scalable inference pipelines, responsible AI and modern generative AI ecosystems.		
	Total Lecture Hours	**45 hours**

Textbook:

Books Details:	**Authors Name**
Deep Learning, MIT Press, 2024	Ian Goodfellow and Yoshua Bengio Aaron Courville.
Neural Networks and Deep Learning, Determination Press, 2015	Michael Nielsen
Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004	S. N. Sivanandam, S. Sumathi, S. N. Deepa

Reference Books:

Books Details:	**Authors Name**
Deep Learning with Python	François Chollet
Advanced Deep Learning with Keras	Rowel Atienza, Neeraj Verma, Valerio Maggio

Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	1. AWS Academy Machine Learning for Natural Language Processing 2. Deep Learning for Developers	30Hrs 15Hrs
Video Lectures	https://www.youtube.com/watch?v=YKulfdsh0b8 https://www.youtube.com/watch?v=aPfkYu_qiF4&list=PLyqSpQzTE6M9gCgajvQbc68Hk_JKGBAY	11Hrs 36Min 80Hrs
Web References	https://aws.amazon.com/what-is/nlp/	5 Hrs

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 (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	-	-	-	-	-	-	-	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, detailed Case study on Agile methodology.

Unit 2	Software Requirement and Quality Assurance	8 hours

Software Requirement Specifications (SRS): Functional and non – functional requirements, 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)
	https://www.youtube.com/watch?v=NlLM3sVF8wY&t=20969s	6(hr)
	https://www.youtube.com/playlist?list=PLZ2ps__7DhBYt5yvXrYAjjWtf5O399Xea	17(hr)
Web References	Software Engineering Tutorial - GeeksforGeeks	

Course Code: BCSCY0613	**Course Name: Malware Analysis and Reverse Engineering**	**L**	**T**	**P**	**C**
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		**3**	**0**	**0**	**3**

Prerequisite: Computer Networks, Operating Systems, C/C++ or Python Programming

Course Objectives: This course introduces techniques for analyzing and understanding malware behavior and reverse engineering malicious software. It equips students with skills to dissect malware through static and dynamic analysis and understand countermeasures and detection techniques.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Demonstrate an understanding of malware, its types, lifecycle, and behavioral characteristics.	K2
CO2	Apply static analysis techniques to examine file structures, metadata, and signatures of malware.	K3
CO3	Perform dynamic analysis to observe runtime behavior and system interaction of malware in isolated environments.	K4
CO4	Detect evasion and obfuscation techniques employed by malware to bypass security mechanisms.	K4
CO5	Develop comprehensive malware analysis reports including IOCs, attack vectors, and mitigation recommendations.	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	3	2	3		3	1		1	3		2	1
CO2	2	3	3	3	3		3	1		1	3	2	3	2
CO3	2	2	3	3	3	1	3	1	1	1	3	3	3	3
CO4	3	2	3	3	3	1	2	2	1	2	2	2	3	3
CO5	3	3	3	2	3	1	3	2	1	2	3	2	2	3

Course Contents / Syllabus

Unit 1	**Introduction to Malware**	**10 hours**

Definition and classification of malware: viruses, worms, Trojans, ransomware, spyware, bots and fileless malware. Malware attack vectors and lifecycle: infection, persistence, execution, command and control (C2), and data exfiltration. Motivation behind malware: cybercrime, hacktivism, cyber espionage, and cyber warfare. Overview of living-off-the-land attacks and script-based malware (PowerShell/macros). Legal and ethical considerations in malware research and responsible handling of malicious software.

Unit 2	**Static Analysis Techniques**	**10 hours**

Analyzing executable file formats: (Windows) and ELF (Linux) structure. Analyzeing executable headers and attributes using tools such as PEStudio, Detect It Easy (DIE), and Exeinfo PE. Extracting and analyzing strings using Strings and FLOSS, File fingerprinting using SHA-256 hashing and reputation analysis with VirusTotal. Detecting and unpacking packed binaries. Introduction to YARA rules for malware identification and classification.

Unit 3	**Dynamic Analysis Techniques**	**10 hours**

Setting up isolated malware analysis environments using VirtualBox/VMware. Monitoring process behaviour using Process Explorer and Process Monitor, Observing network activity using Wireshark, and FakeNet-NG. Identifying system changes: file system, registry, services, and persistence mechanisms. Introduction to automated malware analysis platforms such as Cuckoo Sandbox, Hybrid Analysis or Any. Run. Behavioural indicators and basic malware evasion techniques.

Unit 4	Reverse Engineering Essentials	**8 hours**

Overview of x86 architecture and assembly fundamentals. Understanding program control flow (: loops, branches, function calls) and stack operations. Disassembling binaries using IDA Pro and Ghidra Recognizing common library functions and reconstructing program logic. Debugging malware using x64dbg breakpoints, stepping, memory inspection. Introduction to code obfuscation and anti-disassembly techniques.

Unit 5	Advanced Topics & Reporting	7 hours

Code injection and hooking techniques: DLL injection, and API hooking. Memory forensics using Volatility framework for malware investigation. Anti-analysis techniques: anti-debugging, anti-VM, detection and code obfuscation. Extracting Indicators of Compromise (IOC): IPs, domains, registry keys, file paths, and hashes. Mapping malware behaviour to the MITRE ATT&CK framework. Preparation of professional malware analysis reports including attack summary, indicators and mitigation strategies. Case studies of real-world malware such as WannaCry, Emotet, and Zeus).

Total Lecture Hours	**45 hours**

Textbook:

1. Practical Malware Analysis, Michael Sikorski, Andrew Honig, 1st Edition (2012), No Starch Press.
2. The IDA Pro Book, Chris Eagle, 2nd Edition, No Starch Press.
3. The Art of Memory Forensics, Michael Hale Ligh, Andrew Case, Jamie Levy, Aron Walters, 1st Edition (2014), John Wiley & Sons / Wiley.

Reference Books:

1. Practical Reverse Engineering, Bruce Dang, Alexandre Gazet, Elias Bachaalany, Sebastien Josse, 1st Edition, Wiley.
2. Reversing: Secrets of Reverse Engineering, Eldad Eilam, 1st Edition, Wiley Publishing.
3. Malware Analyst's Cookbook and DVD, Michael Hale Ligh, Steven Adair, Blake Hartstein, Matthew Richard, 1st Edition, Wiley.
4. Learning Malware Analysis, Monnappa K. A., 1st Edition, Packt Publishing.
5. Mastering Malware Analysis, Alexey Kleymenov, Amr Thabet, 1st Edition, Packt Publishing.
6. The Ghidra Book, Chris Eagle, Kara Nance, 1st Edition, No Starch Press.
7. Windows Internals, Part 1 & 2, Pavel Yosifovich, Mark Russinovich, David Solomon, Alex Ionescu, 7th Edition, Microsoft Press.
8. Rootkits and Bootkits, Alex Matrosov, Eugene Rodionov, Sergey Bratus, 1st Edition, No Starch Press.
9. The Shellcoder's Handbook, Chris Anley, John Heasman, Felix Lindner, Gerardo Richarte, 2nd Edition, Wiley.
10. Malware Data Science, Joshua Saxe, Hillary Sanders, 1st Edition, No Starch Press.
11. Windows Kernel Programming, Pavel Yosifovich, 2nd Edition, Leanpub.
12. Rootkits: Subverting the Windows Kernel, Greg Hoglund, James Butler, 1st Edition, Addison-Wesley

Self-Study Content Links:		Duration (in Hrs)
MOOCs	Advanced Malware Analysis - TOC | Infosys Springboard	5 Hrs 3 Min
Video Lectures	1. OpenSecurityTraining.info – Reverse Engineering 2. Malware Unicorn – https://www.malwareunicorn.org/ https://bazaar.abuse.ch/	15 hours 10 hours 10 hours
Web References		

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

Course Code: BCSML0614	**Course Name: Natural Language Processing and Social Media Analytics**	**L**	**T**	**P**	**C**
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		**3**	**0**	**0**	**3**

Pre-requisite: Machine learning, formal languages, Data Structures, Algorithms Probability and Statistics

Course Objectives:

Develop an understanding of the fundamental concepts of Natural Language Processing, apply text vectorization and document similarity techniques, perform text analytics and information extraction, utilize sequential and transformer-based models, and evaluate real-world NLP applications and systems.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand and explain the fundamentals of Natural Language Processing	K2
CO2	Apply text vectorization and similarity techniques.	K3
CO3	Analyze text analytics methods	K4
CO4	Demonstrate knowledge of sequential models and transformer architectures	K3
CO5	Examine real-world NLP 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	3	1	1	2	-	-	-	1	1	-	2	1	1
CO2	3	3	2	2	3	-	-	-	1	1	-	3	2	1
CO3	3	3	2	3	3	1	-	-	1	2	1	3	3	2
CO4	3	2	3	2	3	1	-	-	1	2	1	3	3	2
CO5	3	3	3	2	3	2	1	1	1	2	1	3	3	3

Course Contents / Syllabus

Unit 1 **Introduction to Natural Language Processing** **07 hours**

Definition, Applications and emerging trends in NLP, Challenges. Ambiguity. NLP tasks using NLTK: Tokenization, stemming, lemmatization, stop-word removal, POS tagging, Parsing, Named Entity Recognition, coreference resolution.

Unit 2 **Text Vectorization and Similarity** **10 hours**

Text Vectorization: Bag-of-Words model, N-Gram Models and vector space models, Term Presence, TF-IDF
Textual Similarity: Cosine similarity, Word Mover's distance, Word embeddings: Word2Vec (Skip Gram & CBOW), GloVe.

Unit 3 **Text Analytics** **10 hours**

Text classification, Sentiment Analysis, Topic modelling, LSA, LDA, Opinion Mining, Information Extraction, Information Retrieval.

Unit 4 **Sequential Modelling and Transformers** **10 hours**

Sequential data, Introduction to sequence models - RNN and LSTM, Attention Mechanism, Transformer, Transformer-based models: BERT, FastText, Elmo, GPT, T5, Introduction to Hugging Face Transformers.

Unit 5 **Applications and Case Studies** **08 hours**

NLP applications: Machine translation: Rule-based and statistical approaches, Text summarization Dialog systems, conversational agents and chatbots.
Question Answering and Retrieval-Augmented Generation (RAG): Extractive vs Generative Question Answering, BERT- based QA systems, SQuAD benchmark, Introduction to Retrieval-Augmented Generation (RAG), RAG pipeline: Retriever and Generator components, Evaluation of QA systems, Real-world case studies.

		Total Lecture Hours	**45 hours**
Textbook:			
S.No	**Book Title**	**Authors:**	
1.	Speech and Language Processing, 3rd Edition, Pearson Education, 2008	Daniel Jurafsky, James H.Martin	
2.	Deep Learning for Natural Language Processing, 1st Edition, Apress, 2018	Palash Goyal, Sumit Pandey, Karan Jain	
Reference Books:			
S.no	**Book Title**	**Authors:**	
1.	Natural Language Understanding, 2nd Edition, Pearson, 1995	James Allen	
2.	Neural Network Methods for Natural Language Processing, 1st Edition, Morgan & Claypool Publishers, 2017	Yaov Goldeberg	
Self-Study Content Links:			Duration (in Hrs)
MOOCs	Deep Learning for Natural Language Processing by Prof. Pawan Goyal https://nptel.ac.in/courses/108105572		45Hrs
Video Lectures	https://web.stanford.edu/class/cs224n/?utm		45Hrs
Web References	https://www.nltk.org/?utm https://huggingface.co/docs/transformers/index?utm https://web.stanford.edu/class/cs224n/?utm		30Hrs
Note: Self-study material may include assignments, seminars, micro projects, industrial visits, any other student activities, etc.			

Course Code: BCSE0614	Course Name: WEB DEVELOPMENT USING MEAN STACK	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	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

CO	Course Outcome	Bloom's Level
CO1	Explain, analyze and apply the role of server-side scripting language like Nodejs in the workings of the web and web applications.	K3
CO2	Demonstrate web application framework i.e., Express is to design and implement typical dynamic web pages and interactive web based applications.	K4
CO3	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
CO4	Analyze build and develop single page application using client-side programming i.e. angular js and also develop a static web application.	K4
CO5	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	9 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	9 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	10 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	9 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	45 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

Self Study Content Links:

MOOCs	1. https://www.coursera.org/professional-certificates/ibm-full-stack-cloud-developer?utm_medium=sem&utm_source=gg&utm_campaign=b2c_india_ibm-full-stack-cloud-developer_ibm_ftcof_professional-certificates_cx_dr_bau_gg_sem_pr_in_all_m_hyb_24-06_x&campaignid=21316592778&adgroupid=165695367991&device=c&keyword=courses%20for%20full%20stack%20developer&matchtype=p&network=g&devicemodel=&creativeid=700589190893&assetgroupid=&targetid=kwd-571198547272&extensionid=&placement=&gad_source=1&gad_campaignid=21316592778&gbraid=0AAAADdKX6Y4gVe2nrA-BD75Xjo7XWZvc&gclid=Cj0KCQjw3K7RBhDJARIsAKRtP5T6KxJN_3NMwp5KCs0I8bBY35dSkaUg10Y0FBCfhLqnqqyQCAG9yzgaAonEEALw_wcB 2. https://www.udemy.com/career/full-stack-web-developer/?utm_campaign=Search_Keyword_CareerAccelerator_Prof_la.EN_cc.India&utm_source=google&utm_medium=paid-search&portfolio=CALP-India&utm_audience=mx&utm_tactic=nb&utm_term=_._ag_177463970814_._ad_749449716155_._kw_how%20to%20become%20a%20web%20developer&utm_content=g&funnel=&test=&gad_source=1&gad_campaignid=22487789892&gbraid=0AAAAADROdO0RdwZEytECk3hJ8p0YLN6Wu&gclid=Cj0KCQjw3K7RBhDJARIsAKRtP5SBLzCT5ZsRMLP4tx1IJo-2jQWUMlsGRaHabTFG0lY_G_eZuu7xmCYaAg1tEALw_wcB	40 hrs 30 hrs
Video Lectures	1. https://youtu.be/BLl32FvcdVM 2. https://youtu.be/fCACk9ziarQ 3. https://youtu.be/YSyFSnisip0 4. https://youtu.be/mGVFltBxLKU 5. https://youtu.be/bWaucYA1YRI 6. https://youtu.be/7H_QH9nipNs 7. https://youtu.be/AX1AP83CuK4 8. https://youtu.be/SccSCuHhOw0 9. https://youtu.be/lY6icfhap2o 10. https://youtu.be/z7ikpQCWbtQ 11. https://youtu.be/0LhBvp8qpro	17 hrs

	12. https://youtu.be/k5E2AVpwsko 13. https://youtu.be/SQJkj0WYWOE.list=PLvQjNLQMdagP3OzoBMfBT48uJ-SPfSsWj 14. https://youtu.be/0eWrpsCLMJQ.list=PLC3y8-rFHvwhBRAgFinJR8KHIrCdTkZcZ 15. https://youtu.be/ZSB4JcLLrIo 16. https://youtu.be/0LhBvp8qpro 17. https://youtu.be/k5E2AVpwsko 18. https://youtu.be/SQJkj0WYWOE.list=PLvQjNLQMdagP3OzoBMfBT48uJ-SPfSsWj 19. https://youtu.be/0eWrpsCLMJQ.list=PLC3y8-rFHvwhBRAgFinJR8KHIrCdTkZcZ 20. https://youtu.be/ZSB4JcLLrIo 21. https://youtu.be/Kvb0cHWFkdc 22. https://youtu.be/pQcV5CMara8 23. https://youtu.be/c3Hz1qUUIyQ 24. https://youtu.be/Mfp94RjugWQ 25. https://youtu.be/SyEQLbbSTWg	
Web References	1. https://angular.io 2. https://www.mongodb.com/docs	

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	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.	Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node Paperback, 2nd Edition (2017)	Vasan Subramanian
2.	Hands-On MERN Stack Web Development: Build production-grade web apps and APIs with MERN stack ,3rd Edition (2019)	Shama Hoque
3.	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

Self Study Content Link:		
MOOCs	1. https://www.coursera.org/learn/fullstack-web-development 2. https://www.coursera.org/professional-certificates/ibm-full-stack-cloud-developer	4 Weeks 6 Months
Video Lectures	1. https://youtu.be/CvCiNeLnZ00 2. https://youtu.be/4nKWREmCvsE 3. https://youtu.be/f2EqECiTBL8 4. https://youtu.be/6GQRb4fGvtk 5. https://youtu.be/kTp5xUtcalw	8 hrs 25 Min 7 hrs 8 Min 6 hrs
Web References	1. https://react.dev 2. https://docs.docker.com	

Course Code: BCSCY0612N	Course Name: Blockchain and Cryptocurrency Security	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	0	3

Pre-requisite: Computer Networks, Network Security, Operating Systems, Programming skills

Course Objectives: To explore the foundational concepts of blockchain technology and its architecture to develop secure blockchain-based applications and assess their resilience to cyber threats.

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Understand the evolution, core architecture, types of blockchain, and distinguish between centralized, decentralized and distributed ledger systems and their real-world use cases across IoT, AI/ML, Data Science and other engineering domains.	K2
CO2	Explain the fundamental cryptographic concepts hashing, Merkle trees, digital signatures, wallets and PKI that underpin blockchain security, without requiring advanced mathematical background.	K2
CO3	Apply knowledge of consensus mechanisms, smart contract vulnerabilities and blockchain attack vectors to analyze the security and operational behavior of decentralized networks.	K3
CO4	Develop and deploy smart contracts using Solidity on Remix IDE and understand enterprise blockchain architecture using Hyperledger Fabric, demonstrating ability to write, test and interact with blockchain applications hands-on.	K4
CO5	Evaluate and map real-world blockchain applications across financial systems, supply chain, government, healthcare, IoT and AI/ML domains, and assess emerging trends such as Zero-Knowledge Proofs, Layer 2 solutions and cross-chain interoperability.	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	2	2	2	2	1	2	1	-	-	2	2	-
CO2	3	2	1	2	2	1	1	2	1	1	-	2	1	1
CO3	3	3	2	3	2	2	1	3	2	1	-	3	3	2
CO4	3	3	3	2	3	2	1	3	2	2	-	3	2	2
CO5	2	2	2	2	2	3	2	2	2	2	1	2	2	2

Course Contents / Syllabus

Unit 1	Introduction to Blockchain Technology	08 hours

Evolution and history of blockchain; Structure and components of a blockchain (nodes, blocks, chain, mining); Types of blockchains: Public, Private, Consortium; Centralized traditional databases vs. Decentralized databases vs. Distributed Ledger Technology (DLT), Use cases: cryptocurrency (Bitcoin), supply chain, identity management, healthcare; Digital Money to Distributed Ledgers; Blockchain Design Primitives: Protocols, Permissions (Permissioned vs. Permissionless Blockchains), Privacy
For Demo: https://andersbrownworth.com/blockchain , https://blockchaindemo.io

Unit 2	Cryptographic Foundations	09 hours

Hash functions and Merkle trees (conceptual); Digital signatures and public-key cryptography (how it works, not proofs); Wallets: types and key management; Blockchain addresses and transactions; PKI Infrastructure — conceptual overview; Two-Factor Authentication with Blockchain.
Browser based tool : CyberChef (GCHQ)https://gchq.github.io/CyberChef , https://www.myetherwallet.com

Unit 3	Consensus, Contracts, and Attacks	10 hours

Need for consensus in decentralized systems; Proof-of-Work (PoW) (conceptual); Proof-of-Stake (PoS), Delegated PoS; Proof-of-Authority (PoA); Practical Byzantine Fault Tolerance (PBFT);Mining and its role in blockchain security , Introduction to Bitcoin, Ethereum and altcoins; Transaction lifecycle; UTXO model vs. account-based model; Smart contracts and their working; Common vulnerabilities: reentrancy, overflow/underflow (awareness level); Attacks:51% attack, Sybil Attack, Eclipse Attack, Routing Attack, Distributed Denial of Service (DDoS)
Online Tool: https://www.blockchain.com/explorer, https://txstreet.com/

Unit 4	Enterprise Blockchain and Smart Contract Development	09 hours

Hyperledger Fabric — Components and Architecture; Chaincode: What it is and how it works; Introduction to Remix IDE — Interface and Setup; Writing Your First Smart Contract on Remix IDE; Smart contract development using Solidity; Deploying and Testing Smart Contracts on Remix IDE;Events, Functions and Modifiers in Solidity; ERC-20 Token Standard — Overview , Tools for auditing smart contracts (Mythril, Slither, Oyente) , (Remix IDE)
https://remix.ethereum.org , https://github.com/hyperledger/fabric-samples

Unit 5	Blockchain Applications and Emerging Technologies	09 hours

Introduction of version control engine, use case of automated assignment grading, peer review workflow, public repositories, building portfolios, collaborating on codebases, track changes, and resolve conflicts, Version-controlled technical manuals ensure up-to-date knowledge sharing using GitHub. Regulatory and legal frameworks for cryptocurrencies (global and national perspectives) https://etherscan.io/

	Total Lecture Hours	45 hours

Textbook:

1. Mastering Bitcoin: Programming the Open Blockchain, Andreas M. Antonopoulos and David A. Harding, 3rd Edition, O'Reilly Media, 2023
2. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Daniel Drescher, 1st Edition, Apress, 2017
3. Mastering Ethereum: Building Smart Contracts and DApps, Andreas M. Antonopoulos and Gavin Wood, 1st Edition, O'Reilly Media, 2018

Reference Books:

1. Blockchain Technology and Applications, Pethuru Raj, Kavita Saini and Chellammal Surianarayanan, 1st Edition, CRC Press / Routledge, 2020
2. The Basics of Bitcoins and Blockchains, Antony Lewis, Updated Edition, Mango Media, 2021
3. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, 1st Edition, Princeton University Press, 2016

Self-Study Content Links:		Duration (in Hrs)
MOOCs	1. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012929490529533952761_shared/overview ~ 9 hrs OR 2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013038627363602432360_shared/overview ~ 11 hrs	10 hrs
Video Lectures	1. https://www.youtube.com/watch?v=EH6vE97qIP4 ~2.5 hrs 2. https://www.youtube.com/watch?v=PPFsG92-HiI ~2.5 hrs 3. https://www.youtube.com/watch?v=0UvVOMZqpEA ~1.5 hrs 4. https://www.youtube.com/watch?v=7Cu8KQmUhu0 ~1.5 hrs 5. https://www.youtube.com/watch?v=w7HDA8gUbpQ ~2 hrs 6. https://www.youtube.com/watch?v=JPkgJwJHYSc ~1.5 hrs 7. https://www.youtube.com/watch?v=9mZpc6sKryk ~1.5 hrs 8. https://www.youtube.com/watch?v=RGOj5yH7evk ~1 hr https://www.youtube.com/watch?v=prG1DJMoqoI ~1 hr	15 hours
Web References	1. https://en.wikipedia.org/wiki/Blockchain ~1 hr 2. https://openknowledge.worldbank.org/bitstreams/5166f335-35db-57d7-9c7e-110f7d018f79/download ~1 hr 3. https://arxiv.org/html/2508.01280v1 ~1.5 hrs https://hyperledger-fabric.readthedocs.io/en/latest/whatis.html ~1 hr	5 hours

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

Course Code: BCSML0612	**Course Name: Generative AI**	**L**	**T**	**P**	**C**
Course Offered in: CSE(IoT)/CS		**3**	**0**	**0**	**3**

Pre-requisite: Statistics, Machine Learning, Deep Learning

Course Objectives:

Develop an understanding of deep learning fundamentals and artificial neural networks, design CNN-based computer vision and object detection models, implement sequence learning techniques, apply autoencoders for feature extraction, and build deep learning solutions for real-world AI applications.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Apply the concepts of discriminative vs. generative models.	K3
CO2	Describe GMMs and HMMs via Expectation-Maximization.	K2
CO3	Analyze VAEs with a focus on the reparameterization trick.	K4
CO4	Apply GANs for realistic image generation and training challenges.	K3
CO5	Examine Autoencoders for dimensionality reduction and image reconstruction.	K4

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

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

Course Contents / Syllabus

Unit-1 **Foundations of Generative Model** **9 hours**

Introduction to Generative vs. Discriminative Models, Probabilistic Foundations: Joint, Marginal, and Conditional Distributions, Maximum Likelihood Estimation (MLE) and KL Divergence, Latent Variable Models: GMM, HMM, Use Cases of Generative AI:Image synthesis, Text generation, Healthcare

Unit-2 **Classical Models & Autoencoders** **9 hours**

Gaussian Mixture Models (GMM), Expectation-Maximization, Hidden Markov Models (HMMs)
Autoencoders (AE): Undercomplete, Denoising, Sparse Autoencoders, Variational Autoencoders (VAE): ELBO, Reparameterization Trick, Applications: Image reconstruction, Denoising, Latent Space Interpolation

Unit-3 **Generative Adversarial Networks (GANs)** **9 hours**

GAN Architecture: Generator and Discriminator, Training Challenges: Mode Collapse, Vanishing Gradients, GAN Variants: DCGAN, cGAN, CycleGAN, StyleGAN2, BigGAN, Evaluation Metrics: FID, IS, PR Curves, Applications: Image synthesis, Super- resolution, Image translation, etc.

Unit-4 **Attention Mechanism, Transformers & Large Language Models** **9 hours**

Self-Attention, RNN vs. Attention, Multi-Head Attention, Transformer Architecture: Encoder-Decoder, Positional Encoding, Generative Transformers: GPT (Generative Pretrained Transformer), BERT (Masked Language Model), Fine-Tuning Pretrained Models: GPT-3, T5, BERT, and specialized applications like Text Summarization, Question Answering, LLMs in Action: GPT-3, GPT-4, LLaMA, Prompt Engineering: Understanding and crafting prompts to guide LLM outputs, Applications: Text Generation, Code generation, Conversational AI

Unit-5 **Advanced Generative Models & Fine-Tuning** **9 hours**

Diffusion Models: Forward and Reverse Process, Denoising Diffusion Probabilistic Models (DDPM) Flow-Based Models: NICE, RealNVP, Glow, Fine-Tuning Pretrained Models: GPT-2, Stable Diffusion, Multimodal Models: CLIP, DALL·E, Flamingo, GPT-4 + Multimodal Inputs, Ethical Issues: Deepfakes, AI Bias, Copyright, Content Authenticity, and Responsible AI

Total Lecture Hours **45 hours**

Textbook:		
Book Details		**Authors Name**
Probabilistic Machine Learning: From Algorithms to Applications		Kevin P. Murphy
Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play		David Foster
Hands-On Generative AI with Transformers and Diffusion Models		Omar Sanseviero
Reference Books:		
Book Details:		**Authors Name**
Machine Learning: From the Classics to Deep Networks, Transformers, and Diffusion Models.		Sergios Theodoridis
Essentials of Generative AI		Takeshi Okadome (Springer)
Self-Study Content Links:		**Duration (in Hrs)**
MOOCs	https://onlinecourses.nptel.ac.in/e-learning	24h
Video Lectures	https://youtu.be/vwncYfhxbR0?si=3jjFY5r3nroGNHKK	3h 52m
Web References	https://huggingface.co/learn	15h

LAB Course Code: BCSML0651B	LAB Course Name: Deep Learning Technologies Lab	L	T	P	C
Course Offered in: B. Tech. 3rd Year (6th Semester) Computer Science (CS)		0	0	4	2

Pre-requisite: Python Programming, Machine Learning, Linear Algebra, Probability and Statistics

Course Objectives:

Develop practical skills in designing, training, evaluating, and deploying deep learning models using modern AI frameworks, apply deep learning techniques to solve real-world problems, perform data preprocessing and model optimization, and analyze and improve model performance for efficient and scalable intelligent applications.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Apply deep learning techniques for neural network design, optimization, and computer vision tasks using modern AI frameworks.	K3
CO2	Analyze the performance of intelligent vision models for image classification, object detection, and tracking applications to identify strengths and limitations.	K4
CO3	Evaluate sequence modeling, natural language processing, autoencoders, and generative AI implementations using cloud-enabled AI tools, justifying the choice of techniques for given 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	2	2	2	2	3	1	-	-	3	-	2	2	2	3
CO2	3	2	3	3	3	1	-	-	3	1	2	3	2	3
CO3	3	2	3	3	3	1	-	-	3	1	3	3	2	3

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

Experiment (Case Study Based Question)	CO
A biomedical AI system requires nonlinear learning capability. Implement and compare activation functions: Sigmoid, ReLU, Tanh, and Softmax using Python.	CO1
A business analytics company wants to classify customer behavior patterns. Implement a Multilayer Perceptron (MLP) using TensorFlow/Keras.	CO1
A deep learning model suffers from overfitting. Apply Dropout, Batch Normalization, and Early Stopping techniques to improve performance.	CO1
A cloud AI training pipeline requires optimization benchmarking. Compare SGD, RMSProp, and Adam optimizers for model convergence.	CO1
A surveillance system processes visual inputs. Perform image preprocessing and augmentation using OpenCV/Keras utilities.	CO1
A digit recognition system is required for automation. Train a CNN model using the MNIST dataset.	CO2
A startup wants to classify product images in a scalable AI environment. Design a CNN for image classification on a custom dataset.	CO2
A research team wants to interpret CNN behavior. Visualize feature maps and intermediate activations in CNN models.	CO2
A company wants rapid deployment of vision models. Implement transfer learning using VGG16 or ResNet50.	CO2
A smart city surveillance application requires real-time object detection. Implement object detection using YOLOv8.	CO2
A model must be evaluated for detection accuracy. Calculate IoU, Precision, Recall, and mAP for object detection outputs.	CO2
A surveillance system tracks moving objects in video streams. Implement object tracking using SORT or Deep SORT.	CO2

A customer feedback platform needs sentiment classification. Implement sentiment analysis using NLP preprocessing and deep learning models.	CO3
A chatbot system requires sequence understanding. Implement LSTM/GRU models for sequence prediction or text generation.	CO3
A cloud AI application requires intelligent text generation. Build a prompt engineering or transformer-based generative AI demo using AWS-compatible tools / Hugging Face APIs.	CO3
Multilingual Customer Support Ticket Triage: **Scenario**: A global e-commerce company receives support tickets in English, Spanish, and Hindi. They need to automatically classify tickets into urgent, normal, or low-priority categories based on text, regardless of language. **Tasks:** 1. Preprocess multilingual text (tokenization, stopword removal, language-agnostic vectorization using multilingual embeddings like LaBSE or XLM-R). 2. Implement a Bidirectional RNN or BiLSTM model for sequence classification. 3. Compare performance against a simple feedforward network. Justify why RNN is better for variable-length multilingual sequences. 4. Deploy the model using a cloud NLP service (e.g., AWS Comprehend or Google Translation + AutoML) or local Hugging Face pipeline.	CO3
Anomaly Detection in Sequential Sensor Data **Scenario** A manufacturing plant has sensor readings (temperature, vibration, pressure) over time. Normal sequences follow learned patterns, but anomalies cause early equipment failure. You cannot afford labeled anomaly data for all fault types. **Tasks:** 1. Train a sequence autoencoder (LSTM-based encoder-decoder) on only normal sensor sequences. 2. Use reconstruction error to detect anomalies in test sequences (high error → anomaly). 3. Compare with a sparse autoencoder and a variational autoencoder (VAE) for anomaly scoring. 4. Justify why autoencoder-based approach is chosen over supervised classification.	CO3
Language Model for Medical Discharge Summary Generation **Scenario:** A hospital wants to generate draft discharge summaries from structured data (diagnosis, procedures, medications) to save doctor time. The output must be coherent, medically accurate, and patient-specific. **Tasks** 1. Use a pretrained transformer-based LLM (e.g., GPT-2, BioGPT, or FLAN-T5) fine-tuned on medical discharge data (use Hugging Face or AWS SageMaker JumpStart). 2. Implement prompt engineering to control style and medical terminology. 3. Compare encoder-decoder transformer (T5) vs. decoder-only (GPT) for this text generation task. 4. Evaluate BLEU/ROUGE scores and human readability. **5.** Discuss responsible AI concerns (hallucination, patient privacy) in cloud deployment.	CO3
Multimodal Feature Reduction for Text + Image Search **Scenario:** A social media analytics company needs to reduce high-dimensional text+image feature vectors into a compact representation for fast similarity search. They have paired captions and images. **Tasks:** 1. Extract text embeddings (via BERT) and image embeddings (via ResNet). Concatenate or align them. 2. Train a stacked autoencoder to compress the joint multimodal vector to a lower dimension. 3. Compare with variational autoencoder (VAE) for learning a smooth latent space. 4. Show that compressed representation retains semantic meaning (e.g., similar text-image pairs cluster together).	CO3
Lab 20: Real-Time News Headline Generation Using Sequence-to-Sequence Model **Scenario:** A news aggregator app wants to generate short headlines from long article text in real time. The model must be lightweight enough for cloud scaling but accurate. **Tasks:** **1.** Implement a sequence-to-sequence (seq2seq) model with GRU or LSTM (encoder-decoder with attention) for headline generation. **2.** Compare performance against a pretrained transformer (T5-small) fine-tuned on the same data. **3.** Evaluate on ROUGE scores and inference latency (CPU vs. GPU cloud instances). **4.** Justify which architecture you would deploy for 10,000 requests/second with cost constraints.	CO3
Total Hours: 60 Hrs	

LAB Course Code: BCSE0653Y	LAB Course Name: Software Engineering and Design Lab	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		0	0	4	2

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

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		**Bloom's Level**
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: BCSCY0613P	LAB Course Name: Malware Analysis and Reverse Engineering Lab	L	T	P	C
Course Offered in: CSE(AI)/CSE(AIML)/CSE(DS)/CS/CSE(IOT)/MCT		**0**	**0**	**2**	**1**

Pre-requisite: Computer Networks, Operating Systems, C/C++ or Python Programming

Course Objectives:

This course introduces techniques for analyzing and understanding malware behavior and reverse engineering malicious software. It equips students with skills to dissect malware through static and dynamic analysis and understand countermeasures and detection techniques.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand malware fundamentals, malware categories, attack vectors, executable formats, malware lifecycle, legal and ethical considerations, and basic static malware analysis techniques.	**K2**
CO2	Apply static and dynamic malware analysis techniques using malware analysis tools, sandboxing environments, network monitoring, and behavioral analysis utilities.	**K3**
CO3	Perform reverse engineering, debugging, memory forensics, IOC extraction, anti-analysis detection, and professional malware investigation and reporting.	**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	1	2	3	2	-	2	-	1	-	2	1	2
CO2	2	3	2	3	3	1	-	1	1	1	-	2	1	2
CO3	3	3	3	3	3	2	-	2	2	2	-	2	2	1

List Of Practical's (Indicve & Not Limited To)	**CO**
1. Study malware categories, lifecycle, and attack vectors using real-world malware examples	CO1
2. Analyze legal and ethical issues in malware analysis through case studies	CO1
3. Analyze PE and ELF executable structures using PEStudio and Detect It Easy	CO1
4. Perform string extraction, hashing, and VirusTotal reputation analysis	CO1
5. Detect packed malware and create basic YARA rules for malware identification	CO2
6. Configure an isolated malware analysis environment using VirtualBox/VMware	CO2
7. Monitor malware behaviour using Process Explorer and Process Monitor	CO2
8. Capture and analyze malware network traffic using Wireshark and TCPView	CO2
9. Perform automated malware analysis using Cuckoo Sandbox and observe evasion behaviour	CO2
10. Study x86 assembly instructions and analyze program control flow	CO3
11. Perform binary disassembly and debugging using IDA Pro/Ghidra and x64dbg	CO3
12. Identify obfuscation and anti-disassembly techniques in malware binaries	CO3
13. Perform memory forensics using Volatility and extract Indicators of Compromise (IOCs)	CO3
14. Prepare a professional malware analysis report with mitigation strategies and case study analysis	CO3
Total Hours: - 30 Hours	

LAB Course Code: BCSML0614P	LAB Course Name: Natural Language Processing Social Media Analytics Lab	L	T	P	C
Course Offered in: CS/CSE(IOT)		0	0	2	1

Course Outcome: After completion of the course, the student will be able to		Bloom's Knowledge Level (KL)
CO1	Apply and demonstrate NLP preprocessing techniques using NLTK and spaCy.	K3
CO2	Analyze and justify ML/DL models for text analytics and classification tasks.	K4
CO3	Examine transformer-based NLP applications and chatbots.	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	-	3	1	1
CO2	3	3	2	2	3	1	-	-	1	2	1	3	3	2
CO3	3	3	3	3	3	1	1	1	2	2	1	3	3	3

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

S.No	List of Programs	CO
1	Tokenization of Sentences and Words using NLTK and spaCy	CO1
2	Stemming and Lemmatization on sample text	CO1
3	Stop-word Removal from a document	CO1
4	Part-of-Speech (POS) Tagging of a given sentence	CO1
5	Parsing and Chunking using regex and spaCy	CO1
6	Named Entity Recognition (NER) using spaCy	CO1
7	Coreference Resolution using neuralcoref (or similar)	CO2
8	Bag-of-Words (BoW) vectorization and representation	CO2
9	TF-IDF Implementation and comparison with BoW	CO2
10	N-Gram Model (uni-, bi-, tri-gram) generation from corpus	CO2
11	Cosine Similarity computation between text documents	CO2
12	Word2Vec word embeddings using gensim on a custom corpus	CO2
13	GloVe Embeddings loading and vector representation	CO2
14	Text Similarity using Word Mover's Distance (WMD)	CO2
15	Text Classification using Naïve Bayes/SVM with TF-IDF	CO2
16	Sentiment Analysis using TextBlob and VADER	CO2
17	Topic Modeling using Latent Dirichlet Allocation (LDA)	CO2
18	Topic Modeling using Latent Semantic Analysis (LSA)	CO2
19	Opinion Mining on product/service reviews dataset	CO2
20	Information Extraction (IE) from structured/unstructured documents	CO2
21	Information Retrieval system with ranking using TF-IDF	CO2
22	Sequence Classification using RNN/LSTM (Keras/TensorFlow)	CO2
23	Implementing Attention Mechanism (Basic custom model)	CO2
24	Fine-Tuning BERT for text classification using Hugging Face	CO3
25	Sentence Embeddings using BERT and GPT-2	CO3
26	Text Generation using GPT-2 on a custom dataset	CO3
27	Named Entity Recognition using Transformers (BERT)	CO3
28	Machine Translation using MarianMT or T5 (Hugging Face)	CO3
29	Text Summarization using BART or T5	CO3
30	Chatbot Development using pre-trained transformer (DialoGPT/ChatGPT-style)	CO3
31	Automatic Document Classification and Separation using ML	CO3
32	Hybrid Probabilistic + FSM-Based Sequence Modeling for document boundary detection	CO3

Total Hours: - 30 Hours

LAB Course Code: BCSE0614P	**LAB Course Name: Web Development Using Mean Stack Lab**	**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		**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		**Bloom's Level**
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	-	3	3	2
CO2	3	2	3	2	3	-	-	-	2	2	-	3	3	3
CO3	3	3	3	2	3	-	-	-	1	1	-	3	2	3

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

1. Construct a program to demonstrate usage of Node.js fs Unit to read a file.	CO1
2. Construct a program to display system information using the os Unit.	CO1
3. Construct a program to create and export a simple Node.js Unit.	CO1
4. Construct a program to parse a URL using the Node.js url Unit.	CO1
5. Construct a program to create a basic HTTP server using Node.js	CO1
6. Construct a program to demonstrate callback-based asynchronous execution.	CO1
7. Construct a program to illustrate promise creation and resolution.	CO1
8. Construct a program to demonstrate promise chaining with multiple operations.	CO1
9. Construct a program to show JSON data parsing and stringifying.	CO1
10. Construct a program to demonstrate routing in an Express application.	CO1
11. Construct a program to demonstrate AngularJS data binding.	CO1
12. Construct a program to build a Node.js server that handles GET requests.	CO1
13. Construct a program to implement POST request handling using Express.	CO1
14. Construct a program to perform CRUD operations using Express and MongoDB.	CO1
15. Construct a program to create middleware for logging request details.	CO1
16. Construct a program to serve static files using Express.	CO1
17. Construct a program to create a MongoDB collection and insert documents.	CO1
18. Construct a program to build an AngularJS form with validation.	CO1
19. Construct a program to implement an API call using AngularJS AJAX.	CO1
20. Construct a program to compare synchronous and asynchronous file operations.	CO1
21. Construct a program to demonstrate differences between callbacks and promises.	CO1
22. Construct a program to illustrate REST API endpoints with multiple HTTP methods.	CO1
23. Construct a program to demonstrate middleware execution order in Express.	CO1
24. Construct a program to show one-way and two-way data binding in AngularJS.	CO2
25. Construct a program to analyze MongoDB document structure using Mongoose schema.	CO2
26. Construct a program to measure execution time of synchronous vs asynchronous code.	CO2
27. Construct a program to test REST API endpoints using different HTTP methods.	CO2

28. Construct a program to validate user authentication using Express sessions.	CO2
29. Construct a program to handle error responses with proper HTTP status codes.	CO2
30. Construct a program to demonstrate secure API access using middleware.	CO3
31. Construct a program to develop a RESTful API using Node.js, Express, and MongoDB.	CO3
32. Construct a program to build a single-page application using AngularJS.	CO3
33. Construct a program to integrate AngularJS frontend with Node.js backend.	CO3
34. Construct a program to implement user registration and login system.	CO3
35. Construct a program to build a full-stack application with CRUD operations and API integration.	CO3
Total Hours: 30 hrs	

Course Code: BCS0651	Course Title: Cloud Application and Development	L	T	P	C
Course Offered in: B. Tech. 3rd Year (6th Semester) Computer Science (CS)		0	0	6	3

Pre-requisites:- JDK 1.7/1.8, Eclipse IDE, Dropbox API, Apache tomcat server 7.0/8.0, Google AppEngine API, Servlets, Struts, Spring framework.

Course objective:

Develop the skills to design, deploy, and manage cloud-based applications, apply modern cloud frameworks, platforms, and APIs, and utilize cloud management tools to analyze, optimize, and maintain scalable digital services and cloud computing environments.

Course Outcome: After completion of the course, the student will be able to		**Bloom's Knowledge Level (KL)**
CO1	Understand the concept of cloud computing and the business case for implementing cloud applications in different business scenarios.	K2
CO2	Design cloud-based applications using cloud messaging services like Amazon SQS and RabbitMQ to ensure reliable message delivery and system communication.	K3
CO3	Host cloud-based multiplayer games and construct content delivery networks (CDN) to ensure optimized delivery of media content.	K4
CO4	Automate the management of files, services, and subscriptions in cloud environments using Puppet and Chef.	K5
CO5	Deploy AI models in a cloud environment and evaluate their performance in real-time data processing scenarios.	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	-	-	2	-	-	-	1	1	2	2	2	1
CO2	2	3	3	2	3	-	-	-	2	2	2	3	2	2
CO3	2	3	3	2	3	-	-	-	2	2	2	3	3	2
CO4	2	2	3	2	3	-	-	-	2	2	3	3	3	2
CO5	3	3	3	3	3	1	1	1	2	2	3	3	3	3

Course Contents / Syllabus

UNIT-I	**Core Principles and Application development framework**	**14 Hours**

Business case for implementing cloud application, Requirements collection for cloud application development, Cloud service models and deployment models, Exploring Cloud Access: Comparison Between Web Applications and Cloud-Based Applications, Development Frameworks: MVC Architecture, Struts, and Spring Framework, Industry Cloud Platforms: Google App Engine, Microsoft Azure, OpenShift, and Cloud Foundry

UNIT-II	**Cloud Delivery Models and Architecture Design**	**14 Hours**

Cloud Object Storage and Session Handling, Integration with Third-Party APIs: Understanding Connectivity in Cloud Ecosystems (e.g., Facebook, Twitter, Google APIs), Developing Cloud Applications on AWS: Using Amazon SQS and RabbitMQ for Messaging Services

UNIT-III	**Cloud App Development and Data Management**	**14 Hours**

Amazon SNS, Cloud-Based Multiplayer Game Hosting, and Content Delivery Network Construction, Securing Cloud Data: ACL, OAuth, OpenID, XACML, and Data Transport Security, Ensuring Scalability of Cloud Applications and Services

UNIT-IV	**Automation and provisioning tool**	**15 Hours**

. Puppet and Chef Automation Steps: Overview, Managing Files and Packages, Handling Services and Subscriptions, Executing Commands and Notifications, Using Facts, Conditional Logic, and Logging

UNIT-V	**AI and Machine Learning in Cloud**	**15 Hours**

Cloud AI and ML Services (AWS SageMaker, Google AI, Azure ML), Integrating AI Models into Cloud-Based Applications, Hands-on: Deploying a Simple AI Model in a Cloud Environment, Use Cases in IoT, Real-Time Data Processing, Autonomous Systems, Popular Edge Computing Platforms (AWS Greengrass, Azure IoT Edge)

Total Lecture Hours	**72 hours**

Text books:

1. Mastering cloud computing, TataMcGraw Hill Education Private Limited, 2013, Rajkumar buyya, Christian vecchiola, S Thamarai Selvi
2. Cloud Computing a Practical Approach, Tata McGraw-HILL, 2010 Edition, Anthony T .Velte, Toby J. Velte, Robert Elsenpeter

Reference Books:

1. Cloud computing bible, Wiley publishing, Barrie sosinsky
2. Managing Infrastructure with puppet, O'REILLY , June 2011, James Loope

NPTEL/ Youtube/ Faculty Video Link:

1. https://cloud.google.com/appengine/docs
2. https://www.chef.io/solutions/cloud-management/
3. https://aws.amazon.com/documentation
4. https://www.youtube.com/watch?v=q3m1AB9ECXo
5. https://www.youtube.com/watch?v=q3m1GDFTCPe

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

1. Design and Development of Web applications using MVC Framework
2. Installing and Configuring required platform for Google App Engine
3. Studying the features of GAE PaaS model .
4. Creating and running Web applications (Guest book, MVC) on localhost and deploying the same in Google AppEngine
5. Design and Development of Web applications using Struts.
6. Design and Development of Web applications using Spring framework
7. Developing an ASP.NET based web application on Azure platform
8. Creating an application in Dropbox to store data securely and developing source code using Dropbox API for updating and retrieving files.
9. Installing Cloud Foundry in localhost and exploring CF commands.
10. Cloud application development using open-source virtualization (e.g., VirtualBox, KVM, or Proxmox).
11. Installing and configuring Docker on a local host and running multiple images on a Docker platform.
12. Configuring and deploying VMs/Dockers using Chef/Puppet automation tools.

LAB Course Code: BCSE0659X	LAB Course Name: Applied Research with Mini Project	L	T	P	C
Course Offered in: All Branches		**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.

Course Outcome: After completion of the course, the student will be able to		Bloom's level
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 Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	-	-	1	2	2	1	3	1	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

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. **(CO1)**
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. 9. Incorporate safety, ethics, reliability, sustainability, and security considerations where applicable	Functional prototype/model/system/process with interdisciplinary implementation and practical technical skills. **(CO2)**
Unit 3: Testing, Validation & Optimization	10. Perform testing, experimentation, or validation of developed work. 11. Analyze performance, efficiency, feasibility, or effectiveness.	Validated and optimized project demonstrating performance improvement, technical analysis, and practical feasibility. **(CO2)**

	12. Apply analytical, computational, statistical, or intelligent techniques where relevant. 13. Optimize design, methodology, process, or implementation for improved outcomes	
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. **(CO3)**