ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(AUTONOMOUS)

ACCREDITED BY NBA & NAAC

Affiliated to Andhra University

DEPARTMENT OF INFORMATION TECHNOLOGY

Academic Regulations

Course Structure & Detailed Syllabus (R-23)

III YEAR

Applicable for the batch admitted in 2023-24

IT - COURSE STRUCTURE R23

THIRD YEAR SEMESTER – I											
Code	**Course**	**Category**	**L**	**T**	**P**	**S**	**Total**	**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
23IT6111	Open Elective- I	OE	3	0	0	0	3	40	60	100	3
23IT5111	Professional Elective-1	PE	3	0	0	0	3	40	60	100	3
23PY1101	Fundamentals of Quantum Computing	BS	3	0	0	0	3	40	60	100	3
23IT4120	Artificial Intelligence	PC	3	0	0	0	3	40	60	100	3
23ME3203	Design Thinking	ES	1	0	0	2	0	40	60	100	2
23IT4218	Artificial Intelligence Lab	PC	0	0	3	0	3	50	50	100	1.5
23IT4219	Web Technologies Lab	PC	0	0	3	0	3	50	50	100	1.5
23IT4220	Object Oriented Analysis Design lab	PC	0	0	3	0	3	50	50	100	1.5
23CR9103	Quantitative Aptitude and Effectual Communication Skills	HS	0	0	2	0	2	50	50	100	1
23IT9411	Internship-I	PR	0	0	0	4	3	100	-	100	2
TOTAL			**13**	**0**	**11**	**6**	**26**	**500**	**500**	**1000**	**21.5**

THIRD YEAR SEMESTER – II											
Code	**Course**	**Category**	**L**	**T**	**P**	**S**	**Total**	**Sessional Marks**	**External Marks**	**Total Marks**	**Credits**
23IT6121	Open Elective- II	OE	3	0	0	0	3	40	60	100	3
23IT5121	Professional Elective-2	PE	3	0	0	0	3	40	60	100	3
23IT5131	Professional Elective-3	PE	3	0	0	0	3	40	60	100	3
23IT4121	Design and Analysis of Algorithms	PC	3	0	0	0	3	40	60	100	3
23IT4122	Applied Cryptography	PC	3	0	0	0	3	40	60	100	3
23IT4221	Cryptography LAB	PC	0	0	3	0	3	50	50	100	1.5
23IT5211	Elective Lab – I	PE	0	0	3	0	3	50	60	100	1.5
23IT4221	Internet Of Things Lab	PC	0	0	3	0	3	50	50	100	1.5
23CR9104	High Level Reasoning and Employability Skills	HS	0	0	2	0	2	50	50	100	1
23IT9304	PySpark	SC	0	0	0	2	2	100	-	100	1
TOTAL			**15**	**0**	**11**	**2**	**28**	500	510	**1000**	**21.5**

SEMESETER-1

UNIX NETWORK PROGRAMMING
(PROFESSIONAL ELECTIVE-1)

COURSE CODE: 23IT5111(A) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s):

- C programming
- Basics of Unix systems
- Basics of computer networks

Course Objectives:

- Students will gain the understanding of inter process communication and implementation of different forms of IPC in client-server environment
- Students will gain the understanding of core network programming by using sockets and transport layer protocols like TCP and UDP
- Develop skills in network programming techniques.
- Apply the client server model in networking applications.

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Understand the fundamental concepts of UNIX systems in implementing its IPC.
CO-2: Explain the client-server paradigm and socket structures.
CO-3: Get familiar with the variety of interfaces and frameworks for network applications
CO-4: Apply the applications of sockets and demonstrate skill to design simple applications

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2			2			1			2	1	3	2
CO-2	3	2			3			1			2	2	3	2
CO-3	3	2			3			1			2	2	3	2
CO-4	3	3			1			1			2	2	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **9 Lectures**

Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function.

UNIT II: **9 Lectures**

TCP client server: Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP.

I/O Multiplexing: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server.

UNIT III: **9 Lectures**

Socket options: getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option, ICMPV6 socket option IPV6 socket option and TCP socket options.

Advanced I/O Functions: Introduction, Socket Timeouts, recv and send Functions,readv and write Functions, recvmsg and sendmsg Functions, Ancillary Data, How Much Data Is Queued?, Sockets and Standard I/O, T/TCP: TCP for Transactions.

UNIT IV: **9 Lectures**

Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

Daemon Processes and inetd Superserver: Introduction, syslogd Daemon, syslog Function, daemon_init Function, inetd Daemon, daemon_inetd Function.

Broadcasting: Introduction, Broadcast Addresses, Unicast versus Broadcast, dg_cli Function Using Broadcasting, Race Conditions.

UNIT V: **9 Lectures**

Multicasting: Introduction, Multicast Addresses, Multicasting versus Broadcasting on A LAN, Multicasting on a WAN, Multicast Socket Options, mcast join and Related Functions, dg_cli Function Using Multicasting, Receiving Session Announcements, Sending and Receiving, SNTP: Simple Network Time Protocol, SNTP

Textbooks:

1. W. R. Stevens, B. Fenner, and A. M. Rudoff, *UNIX Network Programming*, 3rd ed. Upper Saddle River, NJ: Pearson Education, 2015.
2. W. R. Stevens, *UNIX Network Programming*, 2nd ed. Upper Saddle River, NJ: Pearson, 2015.

References:

1. T. Chan, *UNIX Systems Programming Using C++*, 1st ed. Upper Saddle River, NJ: Pearson, 2015.
2. G. Glass and K. Ables, *UNIX for Programmers and Users*, 3rd ed. Upper Saddle River, NJ: Pearson Education, 2003.
3. M. J. Rochkind, *Advanced UNIX Programming*, 2nd ed. Upper Saddle River, NJ: Pearson Education, 2004.

e-Resources & other digital material:

1. http://wps.aw.com/aw_kurose_network_2
2. http://www.kohala.com/start/unpv12e.html
3. http://www.netbook.cs.purdue.edu
4. http://www.cs.arizona.edu/llp/book/book.html

Change of syllabus as compared to R-20 Regulations

Unit No	R20 Regulations	R23 Regulations	Changes Incorporated
1	Unix History, Fundamental Concepts, Unix IPC	Introduction to Network Programming, Sockaddr structure, Socket Options	Removed Unix history,IPC. Added Introduction to Network Programming. Sockaddr structure moved to Unit I from Unit II.
2	Overview of Transport Layer Protocols, Sockaddr structure, Socket Options, TCP and UDP client-server examples	Elementary UDP sockets, I/O Multiplexing	Added more focus on UDP sockets. I/O Multiplexing moved to Unit II from Unit III
3	Non-Blocking I/O, Asynchronous I/O, Advanced I/O API	Socket options, Advanced I/O Functions	Added more focus on Socket options, Advanced I/O Functions
4	Overview of Pthreads, Event- driven architecture, Concurrency models for UDP servers.	Elementary name and Address conversions, Daemon Processes and inetd Superserver, Broadcasting	Removed Pthreads, Event- driven architecture. Broadcasting moved to Unit IV from Unit V
5	Broadcasting, Socket creation, design & implementation of ping,trace route.	Multicast Addresses, Multicasting,	Removed implementation of ping,trace route. Added Multicast Addresses and Multicasting
	Overall change: 20%		

Key Modifications:

- **Restructured Topics:** Some topics were shifted across different units.
- **Enhanced Focus on AI Techniques:** More emphasis on Socket options, Advanced I/O Functions, UDP sockets.
- **Removed Topics:** Unix history,IPC ,implementation of ping,trace route, Pthreads, Event- driven architecture
- **Added New Concepts:** Introduction to Network Programming ,Multicast Addresses and Multicasting, Advanced I/O Functions.

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship
Unix Network Programming	23IT5111(A)	R23 AY: 2025-26	• Enhances problem-solving skills for network-based roles. • Develops expertise in socket programming, essential for backend and cloud computing jobs. • Prepares for roles in cybersecurity, with knowledge of TCP/IP vulnerabilities and secure coding practices. • Relevant for system administration and DevOps roles involving server management and automation. • Essential for network engineering roles requiring low-level network troubleshooting and protocol implementation. • Supports AI-driven network automation, anomaly detection, and predictive maintenance in IT infrastructure..	Employability

DATA MINING
(PROFESSIONAL ELECTIVE-1)

COURSE CODE: 23IT5111(B) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s):

- **Overview on Probability and statistics**
- **Knowledge on Database management systems concepts**

Course Objectives:

- To understand the principles of Data Mining.
- To understand the Architecture and concepts of a Data mining system.
- To perform classification, association, clustering and prediction of data.

Course Outcomes:

After completion of the course, the students will be able to:

CO-1: Understanding the basics knowledge on Data mining topics. Apply KDD process for finding interesting pattern from Data. Analyze the kinds of patterns that can be discovered by association rule mining.

CO-2: Evaluate interesting patterns from large amounts of data to analyze for predictions and Classification.

CO-3: Apply clustering techniques and Spatial, Multimedia on real time data.

CO-4: Understand the concepts of Text mining and web data on real time.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	2	3	3	1	1	1				1	3	2
CO-2	3	3	3	3	2	1	1	1				1	3	2
CO-3	3	3	3	2	3	1	1	1				1	3	3
CO-4	3	3	2	3	2	1	1	1				3	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **9 Lectures**

Data Mining: - Introduction, Knowledge Discovery in Databases (KDD), KDD Process, Data Mining Functionalities – Data Preprocessing – Data Cleaning – Data Integration – Data Reduction – Data Transformation and Data Discretization and Concept Hierarchy Generation- Architecture of A Typical Data Mining Systems- Classification Of Data Mining Systems.

Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Apriori Algorithm, FP Growth Algorithm

Learning outcomes: At the end of the unit, the students will be able to

- Understand the basic concepts of Data Mining and KDD process.
- Implement the Data Pre-processing techniques when building new datasets.
- Demonstrate Association Rule mining basket market analysis.

UNIT II: **9 Lectures**

Classification and Prediction: -Basic concepts, Issues Regarding Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Classification by Back propagation – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Model Section.
Knowledge Representation: Ontological Engineering, Categories and Objects, Events.

Learning outcomes: At the end of the unit, the students will be able to

- Apply Classification and Prediction techniques on data to develop systems.
- Utilize Existing Evaluating metrics to measure the Accuracy of a Classifier or Predictor.

UNIT III: **9 Lectures**

Cluster Analysis: - Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods (K-Means) – Hierarchical methods (Agglomerative, /Divisive) – Density-Based Methods (DBSCAN) – Outlier Analysis.

Learning outcomes: At the end of the unit, the students will be able to

- Describe the principles of Cluster Analysis and Outlier Analysis in DM.
- Evaluate different Clustering Methods and their applications in real-world scenarios.

UNIT IV: **9 Lectures**

Mining Object, Spatial, Multimedia: Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining.
Learning outcomes: At the end of the unit, the students will be able to

- Explain the fundamentals of Spatial, Multimedia in DM.
- Construct Data mining Applications according to new challenges in DM.

UNIT V: **9 Lectures**

Introduction to Text mining and web mining: Definition and scope of text mining, Differences between structured, semi-structured, and unstructured data, Applications of text mining in various domains (business, healthcare, social media, etc.). Text Mining on the Web and Social Media. Definition and importance of web mining, Types of web mining. Opinion Mining and Sentiment Analysis on the Web.

Learning outcomes: At the end of the unit, the students will be able to

- Explain the fundamentals of Text mining, web mining w.r.t datamining.
- Utilize Data mining concepts and applications for the benefit of Society.

Textbooks:

1. J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*, 2nd ed. Amsterdam, The Netherlands: Elsevier, 2011.

References:

1. A. Berson and S. J. Smith, *Data Warehousing, Data Mining & OLAP*, 10th Reprint. New Delhi, India: Tata McGraw-Hill, 2007.
2. K. P. Soman, S. Diwakar, and V. Ajay, *Insight into Data Mining Theory and Practice*, Eastern Economy Edition. New Delhi, India: Prentice Hall of India, 2006.
3. G. K. Gupta, *Introduction to Data Mining with Case Studies*, Eastern Economy Edition. New Delhi, India: Prentice Hall of India, 2006.
4. P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Upper Saddle River, NJ: Pearson Education, 2007.

e-Resources:

1. https://nptel.ac.in/courses/106/106/106106093/
2. https://nptel.ac.in/courses/110/107/110107092/
3. https://nptel.ac.in/courses/106/105/106105174/

Activity-Based Learning (Suggested Activities in Class)/ Practical Based learning:

- Flip Class
- Seminar/Poster Presentation
- Role play/Team Demonstration/Collaborative Activity
- Mini Project
- Case study
- Learn by Doing

Change of syllabus as compared to R-20 Regulations

Unit No	R20 Regulations	R23 Regulations	Changes Incorporated
1	In Unit-1 Data warehousing concepts discussed.	Data Mining Introduction, KDD Process, DM Task Primitives, Major issues in DM, Data pre-processing, Application and Trends in DM. Architecture Of A Typical Data Mining Systems	Removed DM Task Primitives. Added Architecture Of A Typical Data Mining System.
2	Introduction to Data Mining. Fundamentals of data mining: Data Mining Functiona1lities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database	Classification and Prediction, Basic concepts, Issues Regarding Classification and Prediction, Decision Tree Introduction, Bayesian Classification, Support Vector Machines, Ensemble Methods	Association Rules discussed in Unit1. In unit 2 added the concepts of Classification and Prediction.

	or Data Warehouse System, Major issues in Data Mining.		
3	Association Rule Problem Definition, Frequent Item Set Generation, APRIORI Principle, FP-Growth Algorithms, MaximalFrequentItemSet, ClosedFrequentItemSet	**Cluster Analysis and its methods**	In R20 Regulation Association Rule discussed in unit 3. Cluster analysis and its methods discussed in unit3.
4	Classification and its concepts.	Mining Object, Spatial, Multimedia, Text and Web Data: Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web- Data Mining Applications- Data Mining and Society.	In unit 4 R20 Classification methods discussed. In unit4 R23 added spatial, multimedia mining methods added.
5	Clustering Overview, ClusteringMethods, partitioningmethods, Key Issues in Hierarchical Clustering, Strengths and Weakness,Outlier Detection.	Text mining and web mining Discussed.	In unit 5 R20 Clustering methods discussed. In unit5 R23 added Text mining and web mining methods added.
	Overall change: 20%		

Key Modifications:

- **Restructured Topics:** Some topics were shifted across different units.
- **Enhanced Focus on Multimedia:** More focus on Multimedia, Spatial and datamining applications,DM for society.
- **Removed Topics:** Primitive DM Task which repeated.
- **Added New Concepts:** Web, Spatial, Multimedia mining methods added.

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship
DWDM	**23IT5111**	R23 AY: 2025-26	• Enhances data mining various on various types of data.. • Develops classification methods for business strategies. • Essential association and clustering methods on data. • Skills relevant for web,spatial data and multimedia, predictive analytics, used in Various domains .	Employability

NETWORK SECURITY
(PROFESSIONAL ELECTIVE-1)

COURSE CODE: 23IT5111(C) CREDITS 3

L T P S Sessional Marks: 40

3 0 0 0 End Exam: 3 Hours

End Exam Marks: 60

Prerequisites: Computer Networks, Operating systems

Course Objectives:

1. To understand basic concepts of network security, including services, attacks, mechanisms, and cryptographic principles.
2. To learn and apply cryptographic techniques, including symmetric and asymmetric encryption, hashing algorithms, digital signatures, and Public Key Infrastructure (PKI).
3. To explore transport and wireless security protocols, such as SSL, TLS, HTTPS, and various wireless standards like WPA and WPA3.
4. To analyse network and system-level threats, and implement security mechanisms such as VPNs, firewalls, and intrusion detection/prevention systems.
5. To examine application-layer security challenges, including email security, IoT security, Network Management security.

Course Outcomes:

Upon successful completion of this course, students will be able to:

CO-1: Describe the fundamental concepts of network security, security mechanisms, and cryptographic techniques. (Unit – 1)

CO-2: Analyze symmetric and asymmetric encryption techniques, hashing algorithms, digital signatures, and the role of Public Key Infrastructure (PKI) in secure communication. (Unit – 2)

CO-3: Evaluate security mechanisms at different network layers, including transport layer (SSL/TLS, HTTPS), wireless security protocols, and network-layer security (VPNs). (Unit – 3 &4)

CO-4: Assess system-level and application-layer security measures, including Intrusion Detection & Prevention Systems (IDPS), firewalls, IoT environments, and network management protocols (SNMP). (Units – 4 & 5)

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	2	2				1						1	2	2
CO-2	2	3	2	2	1							2	3	3
CO-3	2	2	2	2	2	1						1	3	3
CO-4	2	2	2	2		2						2	2	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **8 Lectures**

Introduction to Security, Security Services, Mechanisms and attacks - The OSI Security Architecture- a Model for Network Security - Introduction to Cryptography- encryption and decryption, symmetric and asymmetric key cryptography.

Assignment: Introduction to Crypt Analysis, Strength of Ciphers

Learning outcomes: By the end of this unit, learners will be able to:

1. **Understand fundamental security concepts** – Explain security services, mechanisms, attacks, and the role of the OSI Security Architecture.
2. **Explain cryptographic principles** – Describe the importance of cryptography, including encryption, decryption, and key properties.
3. **Apply encryption techniques** – Demonstrate symmetric and asymmetric cryptographic methods.

UNIT II: **10 Lectures**

Symmetric key ciphers - Block Cipher principles, DES, AES.

Asymmetric key ciphers – Principles of public key cryptosystems, RSA algorithm, Diffie-Hellman Key Exchange.

Hashing Algorithms (SHA, MD5) - Digital Signatures & Certificates - Public Key Infrastructure (PKI)

Learning outcomes: By the end of this unit, learners will be able to:

- **Understand encryption principles** – Explain symmetric and asymmetric encryption, including block and stream ciphers, DES, AES, RSA, and Diffie-Hellman key exchange.
- **Apply cryptographic and authentication techniques** – Demonstrate encryption, hashing (SHA, MD5), and digital signatures for secure communication.
- **Explain the role of Public Key Infrastructure (PKI)** – Describe digital certificates and PKI in ensuring authentication and data integrity.

UNIT III: **10 Lectures**

Network Security Fundamentals: Security at Transport layer: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS

Wireless Network Security: Wireless Transport Layer Security, WAP end-to-end Security, IEEE 802.11i Wireless LAN Security, Wireless Security Protocols: WEP, WPA, WPA2, WPA3.

Learning outcomes: By the end of this unit, learners will be able to:

1. Understand transport layer security – Explain Web security considerations, SSL/TLS, HTTPS.
2. Analyze wireless network security- Describe standards and technologies, including IEEE 802.11i, WAP security, and Wireless TLS, in the context of securing modern wireless communications.
3. Evaluate Wireless Security Protocols -Compare various wireless security protocols (WEP, WPA, WPA2, WPA3), their cryptographic mechanisms, and vulnerabilities.

UNIT IV: **8 Lectures**

Security at **Network layer and** Virtual Private Networks (VPNs)**, System Security: Intrusion Detection & Prevention Systems,** Firewalls & Their Types (Packet Filtering, Stateful, Proxy).

Learning outcomes: By the end of this unit, learners will be able to:

1. Understand network layer security – Explain the role of Virtual Private Networks (VPNs) in securing network communications.
2. Analyse system security mechanisms – Describe Intrusion Detection & Prevention Systems (IDPS) and their role in detecting and preventing cyber threats.
3. Evaluate firewall technologies – Compare different types of firewalls (Packet Filtering, Stateful, Proxy) and their effectiveness in network security.

UNIT V: **9 Lectures**

Security at application layer: Email Security-PGP and S/MIME, Internet of Things (IoT), IoT Security Concepts and Objectives- Overview on Open-Source IoT Security Module.

Network Management Security - Basic Concepts of SNMP- SNMPv1 Community Facility- SNMPv3 enhancements and security features.

Learning outcomes: By the end of this unit, learners will be able to:

1. Understand application layer security – Explain email security mechanisms, including PGP and S/MIME.
2. Analyse IoT Security Concepts - Identify & Explain security requirements, objectives, and challenges of Internet of Things (IoT) environments, including the functionality of open-source IoT security modules.
3. Evaluate the security mechanisms in SNMP protocols - comparing SNMPv1 and SNMPv3 in terms of features and vulnerabilities in network management.

Textbooks:

1. W. Stallings, *Cryptography and Network Security: Principles and Practice*, 8th ed. Upper Saddle River, NJ: Pearson Education, 2023.
2. C. Kaufman, R. Perlman, M. Speciner, and R. Perlner, *Network Security: Private Communication in a Public World*, 3rd ed. Upper Saddle River, NJ: Pearson Education, 2024.

References:

1. W. Stallings, *Network Security Essentials: Applications and Standards*, 4th ed. Pearson Education, 2011.
2. W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 2nd ed. Pearson Education, 2012.
3. D. Stuttard and M. Pinto, *The Web Application Hacker's Handbook*, 2nd ed. Wiley Publishing, Inc., 2011.
4. B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed. John Wiley & Sons, Inc., 2015. ISBN: 0471128457.

Online Resources: Coursera, NPTEL

Change of Syllabus

R-20 Regulation	R-23 Regulation	Changes Incorporated
Unit – I	Unit – I	**Topics ignored:** Mathematical concepts (Euler's Phi-Function, Fermat's Little Theorem, Euler's Theorem, CRT) **Topics included:** OSI Security Architecture. ❖ Shift from theory-heavy math to conceptual security foundation
Unit – II	Unit – II	**Topics ignored:** IDEA, Blowfish, RC4, ECC, Differential & Linear Cryptanalysis **Topics included:** PKI, Digital Signatures, Hashing (SHA, MD5). ❖ Focused more on foundational ciphers and modern applied crypto
Unit – III	Unit – III	**Topics ignored:** Message authentication algorithms, Biometric Authentication, Kerberos, X.509 **Topics included:** SSL/TLS, HTTPS, Wireless TLS, WAP, IEEE 802.11i, WPA3. ❖ Shift from authentication services to transport & wireless security
Unit – IV	Unit – IV	**Topics ignored:** IP Security (AH, ESP). PGP & S/MIME shifted to **Unit V** **Topics included:** VPNs, Firewalls (Packet Filtering, Stateful, Proxy), IDPS. ❖ Added system-level defence mechanisms
Unit – V	Unit – V	**Topics ignored:** Intruders, viruses, password management, SET, Secure E-transactions **Topics included:** IoT Security Concepts, Open-Source IoT Modules, SNMPv1 & SNMPv3. ❖ Emphasis on modern technologies like IoT and network management

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship
Network Security		R23 AY: 2025-26	- Understanding cryptographic algorithms for data security. - Learning authentication protocols and access control mechanisms. - Studying network security threats and mitigation techniques. - Exploring security mechanisms in Cloud, IoT, and Blockchain.	**Employability:** Roles like Security Analyst, Cybersecurity Consultant.

USER EXPERIENCE
(PROFESSIONAL ELECTIVE-1)

COURSE CODE: 23IT5111(D) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): HTML5, CSS, JS

Course Objectives:

- Describe the User Interface.
- Describe the User Experience.
- Learn what the relevant tools are for UX Designers.

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Describe UI and UX design by their own way.

CO-2: Build their different type of prototyping.

CO-3: Fetch all UI Elements in designed page.

CO-4: Design the personas by their own way and Conduct a Usability Test and submit the test Results Report

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	2	2	2	1	3	1	1	1	2	1		1	3	2
CO-2	2	2	2	2	3	2	1	1	2	1		1	3	2
CO-3	2	2	2	2	3	2	1	1	2	1		1	3	2
CO-4	2	2	2	2	3	2	2	1	2	1		2	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: UI/UX Overview (Introduction) 9 Lectures

Introduction – What is UX Design? – What is UI Design? – What is Interaction Design – UX Design Deliverables – Basics of HCI UX Design – User Centered Design – Design Thinking – Activity Based Design – Agile Process – User Research – Competitor Analysis. (TB-1)

Learning outcomes: At the end of this unit the Students will be able to

- Understand the concepts of UI/UX Design and Agile Process.

UNIT II: Interaction Design 9 Lectures

Interaction Design – Ideation Methods – Interaction & Prototyping – Paper Prototyping – Build your own Prototyping – Heuristic (Expert) Evaluation – Designing a Web App.(TB-2)

Learning outcomes: At the end of this unit the Students will be able to

- Understand the concepts of Prototyping and web app design.

UNIT III: Visual Design **9 Lectures**

Visual Design – Web App UI Elements – Grid System – Colors Theory and Palette – Understanding Typography – Material UI and other UI Kit. (TB-2)

Learning outcomes: At the end of this unit the Students will be able to

- Understand the concepts of UI Elements, grid systems and Typography.

UNIT IV: User Research **9 Lectures**

User Research – How to conduct user Interviews – User Research – Creating Personas – Empathy Mapping – Information Architecture – Building User Journey Maps. (TB-3)

Learning outcomes: At the end of this unit the Students will be able to

- Understand the concepts of User Research, Mapping, and Information Architecture.

UNIT V: Usability Testing **9 Lectures**

Usability Testing – Testing Methods – User Testing – A/B testing – Conducting a Usability Test – Test Results Report. (TB-3)

Learning outcomes: At the end of this unit the Students will be able to

- Understand the concepts of Different testing methods and test reports.

Textbooks:

1. P. S. Pyla, *The UX Book: Agile UX Design for a Quality User Experience*, 2nd ed. Cambridge, MA: Morgan Kaufmann, 2019.
2. A. Cooper, R. Reimann, D. Cronin, C. Noessel, J. Csizmadi, and D. LeMoine, *About Face: The Essentials of Interaction Design*, 4th ed. Indianapolis, IN: Wiley, 2014.
3. S. Marsh, *User Research: Improve Product and Service Design and Enhance Your UX Research*, 2nd ed. London, UK: Kogan Page, 2022.

References:

1. A. Boduch, *React Material-UI Cookbook: Build Captivating User Experiences Using React and Material-UI*. Birmingham, UK: Packt Publishing Limited, 2019.

Online Resources:

1. Coursera, "UI/UX Design Specialization." Accessed: Apr. 15, 2025. [Online]. Available: https://www.coursera.org/specializations/ui-ux-design
2. Udemy, "UX Design & User Experience Design Course – Theory Only." Accessed: Apr. 15, 2025. [Online]. Available: https://www.udemy.com/course/how-to-change-careers-and-become-a-uxdesigner/

Change of Syllabus : No changes in R23 Syllabus

3.1.3 of NAAC

Name of the course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/ Skill development/ Entrepreneurship
User Experience	23IT5111	2021-22	1. Web /Mobile app design 2. User Research	Skill development

FUNDAMENTALS OF QUANTUM COMPUTING

COURSE CODE: 23PY3103 CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisites:

- Classical Computing: Binary numbers, Logic gates, Boolean algebra
- Basics of Quantum Mechanics

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Understand the fundamental principles of quantum mechanics relevant to quantum computing.

CO-2: Represent quantum information using qubits and quantum gates.

CO-3: Analyze entanglement and its role in transmitting classical information non-locally.

CO-4: Analyze Bell's theorem and its implications on quantum predictions.

CO-5: Identify practical applications of quantum information theory, such as quantum cryptography

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2	-	-	-	-						2	2	1
CO-2	3	3	2	-	2	-						2	2	2
CO-3	2	3	2	2	2	-						2	2	2
CO-4	2	3	-	2	-	-						2	2	1
CO-5	2	2	2	2	3	1						3	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT - I: Fundamentals of Quantum Mechanics 10 Lectures

Wave Function (ψ) and Schrodinger Equation: Physical significance, Wave particle Duality, One-dimensional potential well: Eigen functions and Eigen values

Why Quantum Computing: Classical vs. Quantum computing, Key applications – (Decoherence, Cryptography and Optimization)

Qubits vs. Classical Bits: Definition and representation, Comparison between bits and qubits

Mathematical Preliminaries: Bra-ket notation, Matrix representation of Quantum States, Inner and Outer products, Tensor product and their significance,

Learning Outcomes:

- Explain the key principles of quantum mechanics relevant to computing.
- Differentiate between classical and quantum computing.
- Understand how quantum states are represented mathematically.

UNIT - II: Single Qubit Quantum Systems **10 Lectures**

Qubit States and Representation: Vector Space and Hilbert Space, Complex vectors and qubits in $\mathbb{C}^2$, Bloch Sphere representation of a Qubit

Single-Qubit Measurement and Operations: Measurement outcomes and probabilities, Basic Quantum gates (**Pauli gates (X, Y, Z)** – Analogous to classical NOT gate, **Hadamard gate (H)** – Creates superposition states, **Phase shift gates (S, T)** – Introduces phase changes **Rotation gates (Rx, Ry, Rz)** – Rotates qubits in Bloch Sphere), Representation of gates using matrices

Learning Outcomes:

- Represent qubits using vector spaces and Bloch Sphere.
- Perform quantum operations using single-qubit gates.
- Understand the mathematical framework of quantum gate operations.

UNIT - III: Multiple Qubit Systems and Entanglement **10 Lectures**

Two-Qubit Systems: Representation and transformation of Two Qubit States, Tensor products and entanglement, The CNOT gate

Entangled States: Visualization and preparation of entangled states, Examples of entangled states in quantum computing

Learning Outcomes:

- Understand how multi-qubit systems interact.
- Analyze quantum entanglement and its role in quantum computing.
- Implement two-qubit gates in quantum circuits.

UNIT - IV: Measurement and Bell's Theorem **10 Lectures**

Quantum Measurement Formalism: Projection operators, Hermitian operators and their significance

Quantum Non-locality and Bell's Theorem: Einstein–Podolsky–Rosen (**EPR**) Paradox and, Bell's inequality and its Significance.

Learning Outcomes:

- Explain quantum measurement and its mathematical formalism.
- Analyze Bell's theorem and its impact on quantum mechanics.
- Evaluate experimental results verifying Bell's inequality.

UNIT - V: Quantum Circuits and Applications **10 Lectures**

Introduction to Quantum Circuits: Operations and representation, Classical vs. Quantum computations, Representation of quantum operations in circuit diagrams

Multi-Qubit Gate Operations: Controlled Gates (CNOT, CZ, Toffoli **)**– Entanglement and logic operations, Universal Gate Sets – How quantum gates can perform universal computation

Applications: Super-dense coding, Quantum teleportation, Quantum cryptography

Learning Outcomes:

- Implement quantum circuits using multi-qubit gate operations.
- Apply quantum principles to real-world problems.
- Understand practical applications like quantum cryptography and teleportation.

Textbooks:

1. M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press.
2. N. David Mermin, *Quantum Computer Science: An Introduction*, Cambridge University Press.
3. Phillip Kaye, Raymond Laflamme, and Michele Mosca, *An Introduction to Quantum Computing*, Oxford University Press.

Reference Books:

1. V. Sahni, *Quantum Computing*, Tata McGraw-Hill.
2. E. Rieffel and W. Polak, *Quantum Computing: A Gentle Introduction*, MIT Press.
3. John Preskill, *Lecture Notes on Quantum Computation* (Available Online).

ARTIFICIAL INTELLIGENCE

COURSE CODE: 23IT4120
L T P S
3 0 0 0

CREDITS 3
Sessional Marks: 40
End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s):

- **Python Programming language**
- **Knowledge on Data Structures**

Course Objectives

- Develop a strong foundation in key principles and theories related to the course subject.
- Gain hands-on experience in applying learned techniques and tools to real-world scenarios.
- Enhance critical thinking and problem-solving abilities by evaluating case studies and implementing solutions.

Course Outcomes

After completion of the course, the students will be able to:

- **CO-1:** Describe the core concepts, principles, and techniques of Artificial Intelligence, including intelligent agents, search algorithms, and knowledge representation. (Units – 1,3)
- **CO-2:** Implement AI problem-solving approaches such as heuristic search, game theory, logical reasoning, planning, and probabilistic reasoning to real-world scenarios. (Units- 1,2,4,5)
- **CO-3:** Design and develop AI-based models using formal logic, constraint satisfaction, planning techniques, and probabilistic methods for intelligent decision-making. (Units – 2,3,4,5)
- **CO-4:** Evaluate AI applications across various domains, considering efficiency, scalability, and ethical implications, while utilizing modern AI tools and frameworks. (Units – 4,5)

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	2	1	1	1	1	1				1	3	2
CO-2	3	3	2	1	2	1	1	1				1	3	2
CO-3	3	3	3	2	2	1	1	1				1	3	3
CO-4	3	3	2	1	2	1	1	1				1	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: Introduction **9 Lectures**

Introduction to AI – AI Definition, The Foundations of AI, The History of AI, The State of the Art

Intelligent Agents – Agents and Environment, The Concept of Rationality, The Nature of Environments, The Structure of Agents

Solving Problems by Searching – Problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies, Heuristic Search Strategies, Heuristic Functions

Learning outcomes: At the end of the unit, the students will be able to

- Explain the fundamental concepts of Artificial Intelligence and the role of intelligent agents in problem-solving.
- Apply various search techniques such as Breadth-First Search, Depth-First Search, and Heuristic Search to solve AI problems.

Unit II: Problem Solving **9 Lectures**

Adversarial Search - Games - Optimal Decisions in Games, Alpha–Beta Pruning

Constraint Satisfaction Problems - Defining Constraint Satisfaction Problems, Constraint Propagation, Backtracking Search for CSPs

Knowledge – Logical Agents – Knowledge-Based Agents, Logic, Propositional Logic, Propositional Theorem Proving

Learning outcomes: At the end of the unit, the students will be able to

- Implement optimal decision-making strategies using Minimax and Alpha-Beta Pruning in game scenarios.
- Solve constraint satisfaction problems using techniques like Backtracking and Constraint Propagation.
- Demonstrate logical reasoning through Propositional Logic, Theorem Proving, and Proof by Resolution.

UNIT III: Knowledge and Reasoning **9 Lectures**

First-Order Logic - Syntax and Semantics of First-Order Logic, Using First Order Logic, Knowledge Engineering in First-Order Logic.

Inference in First-Order Logic - Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution.

Knowledge Representation: Ontological Engineering, Categories and Objects, Events, Reasoning Systems for Categories, Reasoning with Default Information

Learning outcomes: At the end of the unit, the students will be able to

- Apply First-Order Logic for knowledge representation and reasoning in AI systems.
- Develop inference mechanisms using Unification, Forward Chaining, and Backward Chaining techniques.
- Utilize ontological engineering methods for organizing and structuring AI knowledge bases.

UNIT IV: Classical Planning **9 Lectures**

Planning - Definition of Classical Planning, Algorithms for Planning with State Space Search, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches, Time, Schedules, and Resources, Hierarchical Planning.

Learning outcomes: At the end of the unit, the students will be able to

- Describe the principles of Classical and Hierarchical Planning in AI.
- Develop state-space search strategies for planning algorithms and analyze their efficiency.
- Evaluate different planning approaches and their applications in real-world scenarios.

UNIT V: Uncertain Knowledge and Reasoning **9 Lectures**

Quantifying Uncertainty - Acting under Uncertainty, Basic Probability Notation, Bayes' Rule and Its Use

Probabilistic Reasoning - Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Approximate Inference in Bayesian Networks, Relational and First- Order Probability.

Learning outcomes: At the end of the unit, the students will be able to

- Explain the fundamentals of probabilistic reasoning and the application of Bayes' Theorem in AI.
- Construct Bayesian Networks to represent uncertain knowledge and perform approximate inference.
- Utilize probabilistic models for decision-making in AI applications.

Text Books

1. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, NJ: Pearson Education, 2010.

Reference Books

1. E. Rich and K. Knight, *Artificial Intelligence*, 3rd ed. New Delhi, India: Tata McGraw-Hill, 2008.
2. P. H. Winston, *Artificial Intelligence*, 3rd ed. Upper Saddle River, NJ: Pearson Education, 1992.
3. S. Goel, *Artificial Intelligence*. Upper Saddle River, NJ: Pearson Education, 2013.
4. D. W. Patterson, *Artificial Intelligence and Expert Systems*. Upper Saddle River, NJ: Pearson Education, 2015.

e-Resources:

1. https://nptel.ac.in
2. http://aima.cs.berkeley.edu/

Change of syllabus as compared to R-20 Regulations

Unit No	R20 Regulations	R23 Regulations	Changes Incorporated
1	Introduction to AI, History, Intelligent Systems, AI Foundations,	Introduction to AI, Intelligent Agents, Search Algorithms (BFS, DFS, Hill	Removed history, AI languages, and tic-tac-toe game. Added Intelligent Agents and Search Techniques.

	Applications, Tic-Tac-Toe Game, AI Languages	Climbing, Simulated Annealing, Local Search)	
2	State-space search, Exhaustive & Heuristic Search, A*, Constraint Satisfaction, Game Playing, Alpha-Beta Pruning	Optimal Decisions in Games, Alpha–Beta Pruning, Constraint Satisfaction Problems, Knowledge-Based Agents, Propositional Logic, Theorem Proving	Added more focus on propositional logic, inference, and knowledge-based agents.
3	Logic Concepts, Propositional & Predicate Logic, Axiomatic System, Resolution Refutation	First-Order Logic, Knowledge Engineering, Unification, Forward & Backward Chaining, Resolution, Ontological Engineering	Expanded to include Knowledge Engineering and Ontological Engineering
4	Knowledge Representation, Semantic Networks, Frames, Conceptual Dependency, CYC Theory, Semantic Web	Planning, State-Space Search, Planning Graphs, Hierarchical Planning	Knowledge representation moved to Unit III. Unit IV now focuses on AI planning techniques.
5	Expert Systems, Rule-Based Systems, Uncertainty Measures (Bayesian Networks, Dempster-Shafer, Certainty Factors)	Probabilistic Reasoning, Bayesian Networks, Approximate Inference, First-Order Probability	Removed expert systems and added advanced Bayesian methods for handling uncertainty.
	Overall change: 20%		

Key Modifications:

- **Restructured Topics:** Some topics were shifted across different units.
- **Enhanced Focus on AI Techniques:** More emphasis on search algorithms, inference techniques, and probabilistic reasoning.
- **Removed Topics:** Expert Systems and certain Knowledge Representation techniques were removed.
- **Added New Concepts:** AI Planning, Knowledge Engineering, Bayesian Networks, and Advanced Search Algorithms.

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship

Artificial Intelligence	23IT4120	R23 AY: 2025-26	• Enhances analytical thinking and problem-solving skills for AI roles. • Develops logical reasoning for AI-driven business strategies. • Essential for AI applications in automation and robotics. • Skills relevant for AI-based predictive analytics, used in finance, healthcare, and cybersecurity.	Employability

DESIGN THINKING

COURSE CODE: 23ME3203
L T P S
1 0 2 0

CREDITS 2
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisite: Branch specific

Course Objectives:

- To familiarize students with design thinking concepts and principles
- To ensure students can practice the methods, processes and tools of design thinking.
- To ensure students can apply the design thinking approach and have ability to model real world situations.
- To enable students to analyze primary and secondary research in the introduction to design thinking

Course Outcomes

After completion of the course, the students will be able to:

CO-1: Explain the design thinking principles & **Identify** an opportunity and scope of the project and **prepare** the problem statement
CO-2: Apply the empathy tools to study the user and **summarize** finding related to problem for define phase.
CO-3: Describe and **define** the problem specific to the user group and **apply** Ideation tools to **generate** Ideas to **solve** the problem
CO-4: Develop prototypes for test phase.
CO-5: Test the ideas and demonstrate Storytelling ability to present the Ideas.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	2	2	2	2	1	2	1	1	2	2	2	2	2	1
CO-2	2	3	2	3	2	3	1	1	3	2	2	2	2	2
CO-3	2	3	3	2	2	2	1	1	3	2	2	2	3	2
CO-4	2	3	3	2	3	2	1	1	3	2	2	2	3	2
CO-5	2	3	3	3	2	2	1	1	3	3	2	3	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Module 1: Introduction to Design Thinking: Need of design thinking, 7 characteristics of design thinking, comparison of design thinking to other ways of thinking, tools and resources, 5 actions phases of Design thinking, 5 characteristics of action plan. Summary of 5 Thinking mindsets. 5W+H & HMW Tools.

Module 2: Empathy: Think users first, inherent needs of the user, empathize the user, effectively interviewing the users, Ask 5x why, Stake holders Map, Persona, Empathy map,

Module 3: Define: Ask the right question, different types of questions, Design Brief, Opportunity map, POV Statement

Module 4: Ideate: Communicate by drawing, Value of Drawing, rules of ideation, 5 common ideation techniques, Brainstorming, Prioritisation Map, Dot voting, idea evaluation

Module 5: Prototype to Test phase: Types of rough Prototype, need of a Prototype, Need of Prototype testing, Structured Test-Experience lab. Prototype evaluation, observers debrief, Feedback Capture grid

Week	Activity	Marks
1.	Identify an opportunity and scope of the project for providing solution through design thinking.	1
2.	Prepare the initial Problem statement for the identified problem by 5W+H & HMW Tools.	2
3.	Identify the stake holders and prepare the questionnaire to perform Interview for Empathy among stake holders.	2
4.	Apply Ask 5x why tool for identifying the cause identification of the problem.	2
5.	Prepare the Persona based on the responses received from the Stake holders	2
6.	Prepare the Empathy Map/ Customer Journey Map for summarizing pains & gains of stakeholders and insights	2
7.	Prepare the Point of View statement based on user insights and Re-define the problem statement using HMW tool based on the of the customer	2
8.	Perform Brainstorming Session to generate Ideas.	2
9.	Cluster and shortlist the ideas to prepare the prototype	2
10.	Prepare the prototypes for the shortlisted ideas	4
11.	Test the prototype with user and record the responses in feedback capture Grid	2
12	Modify the prototype as per the user feedback.	2

Textbooks:

1. Daniel Ling "Complete Design Thinking Guide for Successful Professionals", Emerge Creatives Group LLP, Print ISBN: 978-981-09-5564-9.
2. Michael Lewrick, Patrick Link, Larry Leifer, The Design Thinking Toolbox, John Wiley & Sons, 2020.

References:

1. Tim Brown, Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation, HarperCollins e-books, 2009.
2. Jeanne Liedtka, Andrew King, And Kevin Bennett, "Solving Problems with Design Thinking" , Columbia University Press Publishers, E-ISBN 978-0-231-53605-9
3. Idris Mootee, "Design Thinking for Strategic Innovation" , 2013 John Wiley & Sons
4. Michael G. Luchs, Scott Swan, Abbie Griffin , "Design Thinking: New Product Development Essentials from the PDMA", ISBN-13 : 978-1118971802
5. Beverly Rudkin Ingle, "Design Thinking for Entrepreneurs and Small Businesses", Apress, ISBN: 9781430261827
6. Jose Betancur "The Art of Design Thinking: Make More of Your Design Thinking Workshops", ISBN: 9781522095378

Web Resources:

- https://dschool.stanford.edu/resources/design-thinking-bootleg
- https://www.ideo.com/post/design-thinking-for-educators
- https://onlinecourses.nptel.ac.in/noc22_mg32/preview
- https://onlinecourses.swayam2.ac.in/imb23_mg65/course

ARTIFICIAL INTELLIGENCE LAB

COURSE CODE: 23IT4218
L T P S
0 0 3 0

CREDITS 1.5
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s):

- **Proficiency in Python**
- **Familiarity with Data Structures & Algorithms**
- **P&S, DMS**

Course Objectives:

- **To introduce AI problem-solving techniques** such as search algorithms, game-playing strategies, and knowledge representation.
- **To develop proficiency in implementing AI algorithms using Python**, including BFS, DFS, Minimax, Bayesian Networks, and Machine Learning models.
- **To enhance logical reasoning and decision-making skills** by working on real-world AI applications like robotics, automation, recommendation systems, and NLP.
- **To explore AI-based planning and probabilistic reasoning techniques**, enabling students to build intelligent agents for uncertainty handling.
- **To provide hands-on exposure to modern AI libraries** like scikit-learn, TensorFlow, pgmpy, networkx, and NLTK.
- **To promote problem-solving and research-oriented thinking** through AI-based projects in domains like healthcare, cybersecurity, and finance.
- **To equip students with industry-relevant AI skills**, preparing them for careers in **Machine Learning, Data Science, and AI Engineering**.

Course Outcomes:

After completion of the course, the students will be able to:

CO-1: Implement fundamental AI problem-solving techniques such as **search algorithms, constraint satisfaction, and game-playing strategies** using Python.
CO-2: Develop AI-based solutions using **knowledge representation, reasoning, and planning techniques** to solve real-world problems in automation, healthcare, and cybersecurity.
CO-3: Apply **probabilistic reasoning and machine learning approaches** to build intelligent systems that handle uncertainty, such as **Bayesian Networks and predictive analytics**.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	2	2	3	1	1	1	-	-	-	2	3	3
CO-2	3	3	3	2	3	2	1	1	-	-	-	2	3	3
CO-3	3	3	3	3	3	2	1	1	-	-	-	3	3	3

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

List of Programs

1. Develop a program to find optimal routes in a graph-based map representation, similar to navigation systems like Google Maps **– CO-1**
2. Write a program to optimize robot path planning in a grid-based environment enabling efficient navigation for AI-powered robotics. **– CO-1**
3. Develop an intelligent decision-making agent for board games such as Chess and Tic-Tac-Toe that can analyze game states and determine optimal moves to compete effectively against human or computer players. **– CO-1**
4. Develop a system to address scheduling and resource allocation challenges in domains such as airline operations, workforce management, or timetable generation by formulating and solving it as a constraint satisfaction problem to ensure feasible and efficient schedules. **– CO-1**
5. Design an AI-based chatbot capable of providing automated assistance in domains such as legal advisory or customer service by performing rule-based decision making and delivering contextually relevant responses. **– CO-2**
6. Develop an AI system for medical diagnosis that models the relationships between diseases and symptoms to enable accurate inference and decision making in clinical scenarios. **– CO-3**
7. Design an intelligent system for automated email classification that can distinguish between spam and legitimate messages by analyzing patterns in message content and metadata. **– CO-3**
8. Develop an intelligent system for disease prediction in healthcare by modeling probabilistic relationships between medical factors such as symptoms, test results, and diagnoses to support clinical decision making. **– CO-3**
9. Design an AI-based system for weather prediction that utilizes probabilistic modeling and inference techniques to generate accurate and reliable forecasts under uncertainty. **– CO-3**
10. Develop an AI-driven solution for predictive maintenance in IoT-enabled systems by modeling uncertainty and analyzing sensor data to forecast potential equipment failures and optimize maintenance schedules. **– CO-3**
11. Develop an AI-based system for automating the process of registering, categorizing, and analyzing First Information Reports (FIRs) to assist law enforcement agencies in streamlining case management, ensuring efficient data retrieval, and supporting decision-making. **– CO-2**

12. Develop an automated fingerprint identification system (AFIS) to efficiently capture, analyze, and match fingerprint data for criminal identification, aiding law enforcement agencies in accurately linking suspects to criminal activities. **– CO-2**
13. Design and implement an expert system, inspired by MYCIN, that uses rule-based inference and knowledge representation techniques to diagnose infectious diseases and recommend appropriate treatments. The system should incorporate a knowledge base, inference engine, and a user interface for interaction, focusing on the application of AI techniques for medical decision support. **– CO-2**

Textbooks:

1. A. Artasanchez and P. Joshi, *Artificial Intelligence with Python: Your Complete Guide to Building Intelligent Apps Using Python 3.x and TensorFlow 2*, 2nd ed. Birmingham, UK: Packt Publishing, 2020.

References:

1. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, NJ: Pearson Education, 2010.
2. T. T. Teoh, *Hands-On Artificial Intelligence with Python*, Springer, 2022

e-Resources:

1. CS50's Introduction to Artificial Intelligence with Python – Harvard (edX)
2. Artificial Intelligence: Search Methods for Problem Solving – University of London (edX)
3. Artificial Intelligence with Python – Udemy
4. Artificial Intelligence for Robotics – Udacity
5. Practical Deep Learning for Coders – Fast.ai
6. AI for Everyone – Andrew Ng (Coursera)

Change of syllabus as compared to R-20 Regulations: New course

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
Artificial Intelligence	23IT4218	R23 AY: 2025-26	• AI Model Implementation in Python → Hands-on learning using scikit-learn, TensorFlow, networkx. • Knowledge Representation & Reasoning → Ontology Engineering, AI-driven Search Systems. • AI for Robotics & Automation → Integration with IoT & Industry 4.0.	Skill Development

WEB TECHNOLOGIES LAB

COURSE CODE: 23IT4219
L T P S
0 0 3 0

CREDITS 1.5
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s): Basic Knowledge on core Java Concepts.

Course Objectives:

- **Understand** the fundamentals of web development, the MERN stack, and the essential tools required for development, such as VS Code, Node.js, MongoDB, and Postman.
- **Develop** structured and responsive web pages using HTML, CSS, Bootstrap, and JavaScript, implementing modern design principles and best practices.
- **Apply** JavaScript concepts, including DOM manipulation, event handling, and ES6 features, to build interactive and dynamic user interfaces.
- **Build** React applications using JSX, components, props, state, React Hooks, and React Router for seamless navigation and efficient state management.
- **Implement** backend development using Node.js and Express.js, creating RESTful APIs, handling middleware, and managing server-side logic.
- **Integrate** MongoDB as a NoSQL database, performing CRUD operations with Mongoose and ensuring secure user authentication with JWT.
- **Connect** the frontend and backend by fetching and managing data in a React application, implementing authentication, and securing APIs.
- **Develop, deploy, and optimize** a full-stack MERN application, ensuring error handling, logging, and performance optimization for production readiness.

Course Outcomes:

After completion of this course, the students will be able to

CO-1: Understand and apply fundamental web development technologies, including HTML, CSS, Bootstrap, JavaScript, and React.js, to build dynamic and responsive user interfaces.

CO-2: Develop and integrate backend services using Node.js, Express.js, and MongoDB, implementing CRUD operations, user authentication, and secure data handling.

CO-3: Optimize, deploy, and maintain full-stack MERN applications by ensuring efficient state management, debugging performance issues, and deploying on platforms like Heroku/Vercel.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2	3		2								3	
CO-2	3	3	3		3								3	3
CO-3	2	3	3		3					2		2		3

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: Frontend Basics

Introduction: Overview of Web Development & **MERN Stack**, Course Structure & Expectations, Development Environment Setup (**VS Code, Node.js, MongoDB, Postman**), Introduction to hypertext markup language (**HTML**), creating web pages, lists, hyperlinks, tables, web forms, inserting images, frames**.**

CSS Fundamentals: CSS Selectors, Box Model, Colors, Fonts, CSS Flexbox & Grid, Responsive Design with Media Queries.

Bootstrap & JavaScript Basics: Bootstrap Grid System & Components, Customizing Bootstrap Styles, JavaScript Basics (Variables, Data Types, Operators, Functions).

Learning outcomes: At the end of this unit the Students will be able to

- **Develop** structured web pages using HTML, incorporating lists, tables, forms, images, and frames.
- **Apply** CSS concepts such as selectors, box model, Flexbox, and Grid to create responsive designs.
- **Utilize** Bootstrap and JavaScript to enhance styling and interactivity in web applications.

UNIT II: React Basics

JavaScript Advanced Concepts: JavaScript DOM Manipulation, Event Handling, ES6 Features (let, const, arrow functions, template literals), Asynchronous JavaScript (Promises, Async/Await).

Introduction to React.js: What is React?, Setting up a React Project using Vite, JSX & React Components, Props & State in React.

React Hooks & Routing: React Hooks (useState, useEffect), Handling Forms in React, React Router for navigation

Learning outcomes: At the end of this unit the Students will be able to

- **Implement** DOM manipulation and event handling using JavaScript and ES6 features.
- **Build** reusable React components using JSX, props, and state management techniques.
- **Integrate** React Hooks and React Router to manage state and enable navigation in single-page applications.

UNIT-3: React Advanced & Backend Basics

State Management & UI Styling: Context API for Global State Management, Private & Public Routes, React Forms & Validation.

Node.js & Express.js: Introduction to Node.js, Setting up an Express.js Server, Middleware & Routing.

MongoDB & Backend Authentication: Introduction to NoSQL & MongoDB, CRUD Operations with MongoDB & Mongoose, User Authentication with JWT.

Learning outcomes: At the end of this unit the Students will be able to

- **Manage** global state in React applications using the Context API and implement form validation.
- **Develop** a backend server using Node.js and Express.js with RESTful API endpoints.

- **Perform** CRUD operations in MongoDB and implement user authentication using JWT.

UNIT IV: Full Stack Development

Connecting Frontend & Backend: Fetching Data from Express API in React, Handling Authentication & Tokens in React.

Learning outcomes: At the end of this unit the Students will be able to

- **Connect** frontend and backend by fetching and managing data from an Express API in a React application.
- **Implement** secure authentication and token-based authorization in a full-stack application.
- **Ensure** seamless data flow and state synchronization between the frontend and backend.

UNIT 5: Full Stack Project & Deployment

Final Project & Deployment: Error Handling & Logging, Deploying MERN App on Vercel/Heroku, Optimizing for Production.

Learning outcomes: At the end of this unit the Students will be able to

- **Develop** a full-stack MERN application following best practices in frontend and backend integration.
- **Implement** error handling, logging, and debugging techniques for application stability.
- **Deploy** the MERN application on cloud platforms like Vercel or Heroku, optimizing it for production.

Textbooks:

1. C. Bates, *Web Programming: Building Internet Applications*, 2nd ed. Chichester, UK: Wiley, 2002.
2. D. Flanagan, *JavaScript: The Definitive Guide*, 7th ed. Sebastopol, CA: O'Reilly Media, 2020.
3. A. Accomazzo, N. Murray, and A. Lerner, *Fullstack React: The Complete Guide to ReactJS and Friends*. [Place not specified]: Fullstack.io, 2017.
4. S. Bradshaw, E. Brazil, and K. Chodorow, *MongoDB: The Definitive Guide*, 3rd ed. Sebastopol, CA: O'Reilly Media, 2019.

References:

1. S. Hoque, *Full-Stack Web Development with Mongo, Express, React, and Node*. Birmingham, UK: Packt Publishing, 2022.
2. M. Casciaro and L. Mammino, *Node.js Design Patterns*, 3rd ed. Birmingham, UK: Packt Publishing, 2020.

List of Programs:

SNO.	PROGRAMS	CO
1	**Week-1:** Program to run a basic HTML file in the browser	1
2	Program to run HTML page with table, forms, images, links, lists, iframe, media.	1
3	**Week-2:** Program to Build a simple personal portfolio page using HTML & CSS.	1
4	Program to Apply Flexbox & Grid for layout.	1
5	**Week-3:** Program to Convert the portfolio page into a Bootstrap-based layout.	1
6	Program to Add a simple interactive form using JavaScript	1
7	**Week-4:** Create a To-Do List app with JavaScript.	2
8	Fetch data from an API using Fetch/Axios.	2
9	**Week-5:** Build a basic React app displaying a list of users from an API.	2
10	Implement state management for user interactions.	2
11	**Week-6:** Add dynamic routing to the React app.	2
12	Build a user registration form.	2
13	**Week-7:** Apply Context API to manage global state.	3
14	Style the React app with Tailwind CSS	3
15	**Week-8:** Build a basic REST API with Express.js	3
16	Test API endpoints using Postman.	3
17	**Week-9:** Connect Express.js with MongoDB.	3
18	Implement JWT authentication for user login.	3
19	**Week-10:** Connect React frontend with Express backend.	4
20	Implement user authentication flow in frontend.	4
21	**Week-11:** Build & deploy a Full-Stack MERN App (Task Manager, Blog, or E-commerce App).	5
22	Prepare a final project presentation.	5

Change of Syllabus: 60% change of Syllabus in R23 Regulation compare to R20 Regulation.

S.No	Existing Topics	New Topics
1	PHP Programming	React.js
2	Django, ZOOMLA, DRUPAL	Node.js & Express.js
3	Content management system (CMS)	MongoDB

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/ Skill development/ Entrepreneurship
Web Technologies Lab	IT319	R15	Creating web pages, JS Forms, Forms API, PHP Forms, CMS Admin (Basics), Site Organization, Creating & Editing Articles, Menus & Modules, Creating Attractive Web Pages with Templates.	Skill Development
Web Technologies Lab	23IT4219	R23	Setting up a React Project using Vite, JSX, Setting up an Express.js Server, Middleware & Routing, Express.js, Introduction to NoSQL & CRUD Operations with MongoDB & Mongoose	Skill Development with Latest Technologies

OBJECT ORIENTED ANALYSIS AND DESIGN LAB

COURSE CODE: 23IT4220
L T P S
0 0 3 0

CREDITS 1.5
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s): Software Engineering, Object-Oriented Programming Language.

Course Objectives:

- Learn the basics of OO analysis and design skills
- Be exposed to the UML design diagrams
- To learn how to write software documents, and communicate with engineers in various forms

Course Outcomes:

After completion of this course, the students will be able to

CO-1: Understand Object-Oriented Concepts

CO-2: Develop and Document Software Design Artifacts

CO-3: Design and Implement UML-Based System Models

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	1	1									1		1	1
CO-2	2	2	2	1				1	1		1	2	1	1
CO-3	3	2	2	1	3			1	1		1	2	1	1

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Study Experiments:

1. A Study on Object oriented model, system and its principle. **CO-1**
2. A study on Object Oriented Analysis: object modeling, dynamic modeling and functional modeling. **CO-1**
3. Overview of SRS **CO-2**
4. High level Design Document (HLD) and Low-Level Design Document **CO-2**

List of Experiments:

1. Brainstorm a software system to develop. **CO-2**
2. Create a Software Requirements Specification (SRS) document for the proposed system. **CO-2**
3. Create HLD for the proposed system **CO-2**
4. Create LLD for the proposed system **CO-2**
5. Develop a Use Case model by identifying and documenting the system's use cases **CO-3**
6. Construct a Domain Model by identifying the system's conceptual classes, and subsequently derive a Class Diagram from this model. **CO-3**
7. To draw the behavioral view diagram: State-chart diagram, Activity diagram **CO-3**
8. To perform the Interaction diagrams: Sequence diagram, Collaboration diagram **CO-3**

9. To perform the implementation view diagram: Component diagram for the system **CO-3**
10. To perform the environmental view diagram: Deployment diagram for the system. **CO-3**
11. Undertake a small-scale project, executing it through the Software Development Life Cycle (SDLC), and generate the necessary UML diagrams for design.

Suggested Case Studies:

1. Passport automation system.
2. Book bank
3. Exam registration
4. Stock maintenance system.
5. Online course reservation system
6. Airline/Railway reservation system
7. Software personnel management system
8. Credit card processing
9. e-book management system
10. Recruitment system
11. Foreign trading system
12. Conference management system
13. BPO management system
14. Library management system
15. Student information system

Suggested Software Tools:

- Open-Source Alternatives: Argo UML, Visual Paradigm
- Eclipse IDE

References:

1. B. Meyer, *Object-Oriented Software Construction*, 2nd ed. New York, NY, USA: Prentice Hall, 1997.
2. G. Booch, J. Rumbaugh, and I. Jacobson, *The UML User Guide*, 2nd ed. Boston, MA, USA: Addison-Wesley, 2005.
3. K. K. Aggarwal and Y. Singh, *Software Engineering*, New Delhi, India: New Age International Publishers, 2005.
4. P. Jalote, *An Integrated Approach to Software Engineering*, 2nd ed. New Delhi, India: Springer, 2005.

Change of Syllabus: New Lab added in R23 Regulation.

1.1.3 of NAAC

Nameof the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
Object Oriented Analyses Design Lab	23IT4220	2025-26	Added model concepts: model, system and principles. Added design concepts: object, dynamic, functional HLLD & LLD UML Model Designing	Skill Development Employability

QUANTITATIVE APTITUDE AND EFFECTUAL COMMUNICATION SKILLS

COURSE CODE: 23CR9103
L T P S
0 0 2 0

CREDITS 1
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisites: Knowledge of LSRW Skills, Basic Maths

Course Outcomes:

After completion of this course, the students will be able to

CO-1: Apply appropriate grammatical structures in written and spoken communication, demonstrating accuracy in voice, speech, and punctuation and apply digital communication strategies in professional environment

CO-2: Develop structured paragraph writing skills and expand vocabulary for precise and contextually appropriate usage in academic and corporate settings and apply critical thinking in writing professional e-mails, proposals, profile structuring, script design and delivery

CO-3: Analyse and interpret reading passages with speed and accuracy, demonstrating proficiency in comprehension, summarization, and fluency an analyse and interpret verbal and nonverbal cues for effective interaction, and making decisions

CO-4: Use their logical thinking and analytical abilities to solve questions from permutations and combinations and geometry and Mensuration and leaning the ability to solve 3D and 2D figures.

CO-5: Understand and solve Tables related questions and analysis the data and solving questions from Data interpretation

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1									2	2		2		
CO-2									2	2		2		
CO-3									2	2		2		
CO-4	2													
CO-5	2													

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

SECTION-A: QUANTITATIVE APTITUDE

UNIT - I: Numerical computation

Vedic Maths, Simplifications, Factorials–progression-Arithmetic Progression, Geometric & Harmonic Progressio

UNIT - II: Modern Maths

Permutations and combinations-linear and circular arrangement, problem based on with and without replacement, repetition and non-repetition Probability –Coin based, Deck Based and Dice problems

UNIT - III: Pure Maths –I

Linear Equations, System of Linear Equations, Quadratic Equations, Representing Linear and Quadratic Equations on co-ordinate system. Surds and Logarithms

UNIT - IV: Pure Maths –II

Geometry- Triangles and its applications, Polygons, Circles, Mensuration-2D and 3D Figures, Co-ordinate Geometry

UNIT - V: Data interpretation – Level - 1 & 2

Bar charts, Pie charts, Graphs. Data interpretation-Bar charts, Pie charts, Tabular graph, Line graph, Combination and Mixed graph. Ratios-proportion, Average in terms of Data interpretation, Data Suffiency in terms Numerical Ability

SECTION-B: EFFECTUAL COMMUNICATION SKILLS

MODULE –I: Transitional Grammar & Communication Skills

UNIT - I:

Transitional Grammar: Active and Passive Voice (knowledge upon the tenses that are applicable); Direct & Indirect speech (importance and usage of the speech); Punctuation (using the signs/symbols appropriately for writing skills)

Listening Comprehension: Reading Skills (Reading Aloud for Stress, Pause and Intonation Training)

UNIT - II:

Paragraph Writing: working on general and specific vocabulary

Presentation Skills: Speech-craft with proper introduction, subject-matter and conclusion

UNIT - III:

Reading Comprehension: Types of passages; Style and tone Strategies for quick and active Reading (importance given to skimming, scanning); Summarizing

Complex Conversations: (in pairs and groups using real life situations); Fillers

UNIT - IV:

Critical Reasoning: Fact, Inference and Judgement (FIJ) Questions

Team-work and its impact of professional work

MODULE –II: Digital Communication and Soft Skills

UNIT - I:

Digital communication: Basics of digital communication, virtual meetings, virtual team management, digital storytelling, presentations using digital platforms

UNIT - II:

Professional Writing: Writing profiles, Business e-mails, proposals, script design and delivery, resume and cover letter, technical documentation

UNIT - III:

Critical Thinking and Decision Making: Critical thinking strategies, Media - Conflict Resolution, leadership and decision making in professional settings

UNIT - IV:

Verbal and Nonverbal Communication: Body language, voice modulation, role-play simulation, clarity, coherence, fluency and barriers of communication

SEMESTER-2

DISTRIBUTED OPERATING SYSTEMS
(PROFESSIONAL ELECTIVE-2)

COURSE CODE: 23IT5121(A) CREDITS 3

L T P S Sessional Marks: 40

3 0 0 0 End Exam: 3 Hours

End Exam Marks: 60

Prerequisite(s): Operating Systems, Computer Networks

Course Objectives:

- To introduce the foundations of Distributed Systems.
- Introduce the idea of peer-to-peer services and distributed file system.
- Examine in detail the system level and support required for distributed system.
- Discover the issues involved in studying distributed process and resource management.

Course Outcomes:

CO-1: Apply the critical operations involved in designing and establishing the communication in distributed systems.

CO-2: Examine the models used to implement a consistent distributed Shared Memory system which also handles clock synchronization and deadlocks.

CO-3: Evaluate the methods of Process and Resource Management to balance and share the load in distributed system.

CO-4: Make use of distributed File System and Naming mechanisms for accessing, sharing and naming the files in distributed systems and its related applications, and Understand the importance of ongoing security monitoring in the Distributed Systems

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2	1			1	1	1	1		1	2	3	2
CO-2	2	3	2	1		1	1	1	1	1	1	2	3	2
CO-3	3	2	1	1		1	1	1	1	1	1	2	3	2
CO-4	2	2	2	3		1	1	1	1	1	1	2	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: Fundamentals of DOS and Message passing 10 Lectures

Fundamentals: Distributed computing system, models, Architecture of DS, Distributed operating system, design issues, introduction to DCE.

Message passing: Introduction, features, issues in IPC, synchronization, Buffering, multi datagram messages, encoding and decoding, process addressing, failure handling, group communication,

Learning Outcomes: At the end of this unit, the students will be able to

- Understand the fundamentals used in designing a distributed system.
- Identify how message passing takes place in distributed systems..

UNIT II: Remote Procedure Call and Distributed Shared Memory 10 Lectures

Remote procedure call: Introduction, RPC model, Transparency of RPC, Implementation, Stub generation, RPC messages, server management, parameter-passing semantics, call semantics, communication protocols, complicated RPC's, Client-Server

Binding, exception handling, security, lightweight RPC
Distributed Shared memory: Introduction, general architecture, design and implementation issues, granularity, structure of shared memory space, consistency models, replacement strategy, thrashing

Learning Outcomes: At the end of this unit, the students will be able to

- Analyze the working of RPC in distributed systems
- Design a consistent and distributed shared memory model.

UNIT III: Synchronization and Resource Management 10 Lectures

Synchronization: Introduction, Clock Synchronization, Event ordering, Mutual Exclusion, Deadlock, Election Algorithms
Resource management: Introduction, Desirable Features of a good global scheduling algorithm, Task assignment approach, load-balancing approach, load-sharing approach.
Process Management: Introduction, process migration, Threads.

Learning Outcomes: At the end of this unit, the students will be able to

- Experiment with clock synchronization and handling deadlocks in distributed systems
- Examine the different load-balancing, load-sharing approaches.
- Organize processes and threads.

UNIT IV: Distributed File Systems 10 Lectures

Distributed file systems: Introduction, features, file models, Accessing models, sharing models, file- caching schemes, file Replication, Fault tolerance, Atomic transactions, design principles. Consensus Algorithms – Proof of Work, Proof of Stake

Learning Outcomes: At the end of this unit, the students will be able to

- Design a distributed file system.
- Make use of different naming mechanisms.

UNIT V: Naming, Vulnerabilities and Threats 10 Lectures

Naming: Introduction, features, fundamental terminologies, system-oriented names, object- locating mechanisms, human-oriented names, name caches, naming and security

Vulnerabilities and Threats in Distributed Systems: Importance, Impact of Security Breaches, Common Vulnerabilities - Weak Authentication and Authorization, Software Bugs and Flaws, Lack of Monitoring and Logging, Poor Data Protection, Threats to Distributed Systems - Denial of Service (DoS) Attacks, Data Interception and Eavesdropping, Insider Threats, Insider Threats

Learning Outcomes: At the end of this unit, the students will be able to

- Understand the importance of naming in distributed systems
- Learn methods to secure names and naming systems, such as encryption, integrity checking, and access control mechanisms.

Textbooks:

1. P. K. Sinha, *Distributed Operating Systems: Concepts and Design*, New Delhi, India: Prentice Hall of India, 2007.

References:

1. A. S. Tanenbaum and M. Van Steen, *Distributed Systems: Principles and Paradigms*, 2nd ed. Boston, MA, USA: Pearson Education, 2007.
2. M. L. Liu, *Distributed Computing: Principles and Applications*, Boston, MA, USA: Pearson Education, 2004.
3. N. A. Lynch, *Distributed Algorithms*, San Francisco, CA, USA: Morgan Kaufmann Publishers, 2003.

Change of Syllabus

(R20) Syllabus	**(R23) Syllabus**
In (R20) DOS subject code is IT322	In (R23) DOS subject code is 23IT5121(A)
UNIT-I, II & III	Removing Basics of Computer Networking concepts Remaining concepts interchanged between the units
UNIT-IV	Added Consensus Algorithms,CAP Theorm
UNIT-V	Added Security concepts and real time case-studieslike GFS,Bitcoin,Google Cloud Store
Change of Syllabus	20%

Note: This course is related to Employability/Skill development.

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/ Skill development/ Entrepreneurship**
Distributed Operating Systems	23IT5121(A)	2015-2016	Design issues, Internetworking, group communication, Implementation, Client- Server Binding, structure of shared memory space, consistency models, Mutual Exclusion, Deadlock, Election Algorithms, load-balancing approach, load-sharing approach, file-caching schemes and system-oriented names, human-oriented names and naming and Security and case studies.	**Employability**

MACHINE LEARNING
(PROFESSIONAL ELECTIVE-2)

COURSE CODE: 23IT5121(B) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): Probability, Linear Algebra, Python Programming

Course Objectives:

- To give basic knowledge about the machine learning models and theory that forms the foundation of machine learning.
- Identify and apply the appropriate Machine learning technique to classification, tree models, rule models, probabilistic models and ensemble techniques.

Course Outcomes:

After completion of the course, the students will be able to:

CO-1: Define machine learning, types of ML, describe the extraction of data, and fine tune the data for applying ML model.

CO-2: Analyze the data and predict decisions using rule and linear classifier models.

CO-3: Classify the data by using distance-based and probabilistic models.

CO-4: Explore the feature transformations and ensemble techniques.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2				1		1	1	1		1	3	2
CO-2	2	3	2					1	1	1		1	3	2
CO-3	3	2						1	1	1		1	3	2
CO-4	2	3	3		2	2	2	1	1	1		1	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **9 Lectures**

Tasks: the problems that can be solved with machine learning

Models: The output of machine learning: Geometric models, Probabilistic models, Logical models, Grouping and grading

Features: The workhorses of machine learning: Two uses of features, Feature construction and transformation, Interaction between features

Learning outcomes: At the end of this unit, the students will be able to

- Understand features and tasks can be performed by machine learning.
- Selection, Train a model and Fine tuning the model for real data .

UNIT II: **9 Lectures**

Linear Regression, Gradient Descent, Polynomial Regression, Logistic Regression.

Training a Binary Classifier, Performance Measures, Multiclass Classification, Error Analysis, Multilabel Classification, Multioutput Classification.

Learning Outcomes: At the end of this unit, the students will be able to

- Explore the Regression models.
- Understand the process of training a binary classifier

UNIT III: **9 Lectures**

Rule models: Learning ordered rule lists, Learning unordered rule sets, Descriptive rule learning, First-order rule learning

Linear models: The least-squares method, The perceptron, Support vector machines, Obtaining probabilities from linear classifiers, Going beyond linearity with kernel methods

Learning outcomes: At the end of this unit, the students will be able to

- Explain rule models
- Explore liner models

UNIT IV: **9 Lectures**

Distance-based models: Neighbours and exemplars, Nearest-neighbour classification, Distance- based clustering, Hierarchical clustering, From kernels to distances

Probabilistic models: The normal distribution and its geometric interpretations, Probabilistic models for categorical data, Discriminative learning by optimizing conditional likelihood, Probabilistic models with hidden variables, Compression-based models

Learning outcomes: At the end of this unit, the students will be able to

- Describe neighbourhood classifier models
- Explain distributions and probabilistic models

UNIT V: **9 Lectures**

Features: Kinds of features, Feature transformations, Feature construction and selection **Model ensembles:** Bagging and random forests, Boosting, Mapping the ensemble landscape

Learning outcomes: At the end of this unit, the students will be able to

- Describe Features
- Explain various ensembling models.

Textbooks:

1. P. Flach, *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*, Cambridge, U.K.: Cambridge University Press, 2012.

References:

1. E. Alpaydin, *Introduction to Machine Learning*, 2nd ed. New Delhi, India: PHI Learning, 2010.
2. P. Baldi and S. Brunak, *Bioinformatics: A Machine Learning Approach*, Cambridge, MA, USA: MIT Press, 2002.
3. M. Kearns and U. Vazirani, *An Introduction to Computational Learning Theory*, Cambridge, MA, USA: MIT Press, 1994.

4. T. M. Mitchell, *Machine Learning*, New York, NY, USA: McGraw-Hill, 1997.

Change of syllabus: No change

1.1.3 of NAAC

Name of the Course	**Course Course**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
Machine Learning	23IT5121	R-20	Working with Real Data, Training a Binary Classifier, The perceptron, Support vector machines	Skill development

CYBER SECURITY
(PROFESSIONAL ELECTIVE-2)

COURSE CODE: 23IT5121(C) CREDITS 3

L T P S Sessional Marks: 40

3 0 0 0 End Exam: 3 Hours

End Exam Marks: 60

Prerequisite(s):

- Computer Networks
- Network Security

Course Objectives:

- Understand cyber security principles, threats, and defence mechanisms to protect digital systems.
- Analyze cybercrime techniques, vulnerabilities, and risk management strategies in various computing environments.
- Apply security frameworks, cryptographic techniques, and access control mechanisms in networks, databases, and cloud systems.
- Develop skills in cyber threat intelligence, forensic investigation, and incident response for ensuring compliance and resilience.

Course Outcomes:

After course completion, the students will be able to:

CO-1: Analyze key principles, challenges, and threat landscapes, including human factors and the cybersecurity kill chain.

CO-2: Recognize categories of cybercrimes, analyze vulnerabilities in software and systems, and implement defense mechanisms.

CO-3: Apply security frameworks, risk management strategies, and protection techniques in enterprise infrastructure.

CO-4: Leverage threat intelligence, conduct forensic analysis, and manage cybersecurity incident response and recovery.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	1	1	2	2	3	2	2	1	2	2	3	2	3
CO-2	2	2	3	2	2	2	2	1	1	2	3	3	3	2
CO-3	3	1	2	2	3	3	2	1	2	2	3	2	2	3
CO-4	2	2	3	2	3	3	2	1	1	2	2	2	3	3

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **10 Lectures**

Cybersecurity: Definition, principles.

Cybersecurity Challenges: Shift in the threat landscape.

Cybercrime: Categories of cybercrime.

Tools and Methods Used in Cybercrime: Phishing, password cracking, keyloggers, viruses

and worms, DDoS attacks, SQL injection, buffer overflow.

Understanding the Cybersecurity Kill Chain: External reconnaissance, access, privilege escalation.

Authentication, Authorization, and Accountability (AAA): Access control, identity management, user authentication, technical aspects of accountability.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Explain fundamental cybersecurity principles and evolving cyber threats.
2. Identify various cybercrimes and their methods of execution.
3. Analyze and apply cybersecurity frameworks such as the kill chain model.

UNIT II: 10 Lectures

Software Security: Categories of vulnerabilities, prevention and detection of vulnerabilities, mitigating exploitation of vulnerabilities.

Security in the Design of Operating Systems: Simplicity of design, layered design, kernelized design, reference monitor – correctness and completeness, secure design principles, trusted systems, trusted system functions.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Identify software vulnerabilities and employ secure coding practices.
2. Analyze security models used in operating systems.

UNIT III: 10 Lectures

Web and Mobile Security: Fundamental concepts and approaches, sandboxing, client-side and server-side vulnerabilities and their mitigations.

Cybercrime: Mobile and Wireless Devices: Proliferation of mobile and wireless devices, trends in mobility, Online Frauds in mobile and wireless computing era, security challenges posed by mobile devices, registry settings for mobile devices, authentication service security, attacks on mobile phones.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Identify and mitigate vulnerabilities in web applications.
2. Recognize cybersecurity risks in mobile and wireless environments.
3. Apply secure authentication practices for mobile and wireless security.

UNIT IV: 10 Lectures

Database Security: Security requirements of databases, reliability and integrity, database disclosure.

Cloud Computing Security: Introduction to cloud computing, service and deployment models, risk analysis, Security control in cloud, cloud security tools and techniques, cloud identity management, securing IaaS.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Analyze security threats in databases and implement security mechanisms.
2. Understand cloud security principles and risk mitigation techniques..

UNIT V: **10 Lectures**

Threat Intelligence: Introduction, open-source tools, Threat Intelligence, leveraging threat intelligence to investigate suspicious activity.

Investigating an Incident: Investigating an incident, scoping the issue, key artifacts, investigating a compromised system on-premises, investigating a compromised system in a hybrid cloud.

Recovery Process: Disaster recovery planning process, challenges.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Utilize threat intelligence tools to analyze cyber threats.
2. Conduct digital forensics investigations and implement incident response strategies.

Textbooks:

1. N. Godbole and S. Belapure, *Cyber Security: Understanding Cybercrimes, Computer Forensics, and Legal Perspectives*, 1st ed. New Delhi, India: Wiley India, 2011.
2. C. P. Pfleeger, S. L. Pfleeger, and J. Margulies, *Security in Computing*, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2018.

References:

1. A. Rashid, H. Chivers, G. Danezis, E. Lupu, and A. Martin, *The Cyber Security Body of Knowledge*, 1st ed. U.K., 2019.
2. Y. Diogenes and E. Ozkaya, *Cybersecurity – Attack and Defense Strategies*, 3rd ed. Birmingham, U.K.: Packt Publishing, 2022.
3. C. J. Brooks, C. Grow, P. Craig, and D. Short, *Cybersecurity Essentials*. Hoboken, NJ, USA: John Wiley & Sons, Sybex (A Wiley Brand), 2018.
4. C. McNab, *Network Security Assessment*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, Inc., 2016.
5. W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 2nd ed. Boston, MA, USA: Pearson Education, 2013.
6. W. Stallings, *Network Security Essentials: Applications and Standards*, 4th ed. Boston, MA, USA: Pearson Education, 2011.

Web References:

1. OWASP - Open Web Application Security Project: https://owasp.org
2. NIST Cybersecurity Framework: https://www.nist.gov/cyberframework
3. SANS Institute: https://www.sans.org/
4. CIS - Center for Internet Security: https://www.cisecurity.org
5. ISACA: https://www.isaca.org

Change of Syllabus: New Course

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship
Cyber Security	23IT5121 (C)	R23 (2025)	Cybersecurity Fundamentals (Threats, Kill Chain, Attack Strategies), Secure Coding & Software Security, Web, Mobile, and Wireless Security (Malware, Phishing, Network Attacks), Cloud & Database Security, Cyber Threat Intelligence, Digital Forensics, and Incident Response.	Employability

SOFTWARE DESIGN PATTERNS
(PROFESSIONAL ELECTIVE-2)

COURSE CODE: 23IT5121(D) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): Software Engineering, OOAD with UML

Course Objectives:

- Understand the creational and structural patterns.
- Create a solution and design for a given problem.
- Create multiple solutions for a given problem.
- Be able to identify different structural and behavioural patterns.

Course Outcomes:

After course completion, the students will be able to:

CO-1: Demonstrate an understanding of a range of design patterns.
CO-2: Experience core design principles and to assess the quality of a design with respect to these principles.
CO-3: Capable of applying these principles in the design of object oriented systems.
CO-4: Identify and apply suitable patterns in specific contexts

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	2				1		2				3	3
CO-2	3	3		2				1					3	3
CO-3			3	2		2			3					3
CO-4	3	3			2				1					2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **9 Lectures**

Design Patterns: Pattern Description, Organizing catalogs, role in solving design problems, Selection and usage.

Creational Patterns: Abstract factory, Builder, Factory method, Prototype, Singleton, Benefits of Creational design patterns.

Learning Outcomes: student will be able to

- Understand what is the use of design patterns.
- Analyse how problems can be solved using design patterns.
- Analyse how to use creational design patterns for different problems.
- Evaluate the product design can be independent or not.

UNIT II: **9 Lectures**

Design Pattern Catalog: Structural Patterns**,** Adapter, Bridge, Composite, Decorator, Façade, Flyweight, PROXY.

Learning Outcomes: student will be able to

- Understand the structure of design patterns.
- Analyse how classes and objects are composed to form large structures.
- Understands the logic of design patterns.

UNIT III: **9 Lectures**

Behavioural Patterns : Chain of responsibility, command, Interpreter, state, strategy, template method, Visitor.

The World Wide Web - a case study in Interoperability,

Air Traffic Control – a case study in designing for high availability,

Celsius Tech – a case study in product line development.

Learning Outcomes: student will be able to

- Analyse responsibilities between objects of classes.
- Apply complex control flow in applications.
- Analyse various dependencies between objects with respect to classes.

UNIT IV: **10 Lectures**

Envisioning Architecture: The Architecture Business Cycle, Overview of Software Architecture

Learning Outcomes: student will be able to

- Understand the necessity of Architecture Business Cycle and importance of Software Architecture.
- Understand software architecture with various reference models.
- Understand documenting software architectures.
- Analyse the Quality Attributes.

UNIT V: **10 Lectures**

Creating Architecture: Understanding Quality Attributes, Achieving Qualities, Designing the Architecture

Air Traffic Control: A Case Study in Designing for High Availability

Flight Simulation: A Case Study in an Architecture for Integrability

Learning Outcomes: student will be able to

- Understand Architecture Evaluation.
- Understand different analysis models.
- Analyse Architecture design decision making.
- Analyse how software architectures can used in different software applications.

Textbooks:

1. E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Boston, MA, USA: Pearson Education, 1995.
2. L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*, 2nd ed. Boston, MA, USA: Addison-Wesley, 2003.

References:

1. Luke Hohmann , Beyond Software architecture, Addison wesley, 2003.

2. David M. Dikel, David Kane and James R. Wilson, Software architecture, 1st Edition, Prentice Hall,2001
3. F.Buschmann , Pattern Oriented Software Architecture, Wiley&Sons,1st Edition,2001
4. Brahma Dathan, Sarnath Rammath, *Object-oriented analysis, design and implementation*, Universities Press, 2013.

Change of Syllabus: No change

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
SOFTWARE DESIGN PATTERNS	23IT5111 (D)	R23 (2024)	Software Architecture, Architectural patterns, reference models, ATAM, CBAM , Prototype, Singleton, Structural Patterns ,Adapter, Bridge,Proxy.	Employability/Skill development

MOBILE ADHOC NETWORKS
(PROFESSIONAL ELECTIVE-3)

COURSE CODE: 23IT5131(A) CREDITS 3

L T P S Sessional Marks: 40

3 0 0 0 End Exam: 3 Hours

End Exam Marks: 60

Prerequisite(s): Computer Networks

Course Objectives:

- To give an understanding of the basic knowledge on wireless LANs, adhoc wireless networks, and protocols.
- To give an overview of networking principles and how the wireless protocols, routing, operate.
- To know the basic background in wireless networks that will allow them to practice in this field and that will form the foundation for more advanced courses in networking.
- To acquire the basic skills needed to write network applications in software tools i.e Netsim, NS2/NS3.
- To give an overview of the issues and challenges in adhoc networks.

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Develop an understanding of wireless communication systems, including Wireless LANs, PANs, and Adhoc networks, and explore practical applications for creating localized networks.

CO-2: Acquire knowledge of network management principles, focusing on MAC protocols and routing strategies for efficient data transmission in dynamic environments.

CO-3: Evaluate Quality of Service (QoS) parameters and analyze various energy management schemes to enhance network performance.

CO-4: Assess the significance of energy management in network operations, proposing sustainable solutions for optimizing resource utilization.

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	2	3	2	1	3						3	1	2	1
CO-2	3	1	1											1
CO-3	3	2	1	2		3					2	1	1	1
CO-4	3					2	1				2	1	1	

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **8 Lectures**

Wireless LANS AND PANS: Introduction Fundamentals of WLANS, IEEE 802.11 Standard, HIPERLAN Standard, Bluetooth, HomeRF.

Wireless Internet: Wireless Internet, Mobile IP, TCP in wireless Domain WAP, Optimizing Web over Wireless.

Learning outcomes: At the end of this unit, the students will be able to

- Learn different Wireless local area networks and personal area networks.
- Learn the mobile internet protocol.

UNIT II: **10 Lectures**

Adhoc Wireless Networks: Introduction in Ad-Hoc wireless Networks, Ad-Hoc Wireless internet.

MAC Protocols For Adhoc Wireless Networks: Introduction, issues in designing a MAC protocol for adhoc wireless networks, design goals, classification of MAC protocols, contention based protocols.

Learning outcomes: At the end of this unit, the students will be able to

- Learn basics of wireless ad hoc networks
- Learn the different MAC protocols in wireless adhoc networks

UNIT III: **12 Lectures**

Routing Protocols: Introduction, issues in designing a routing protocol for adhoc wireless networks, classification, table-driven routing protocols, on-demand routing protocols, hybrid routing protocols, hierarchical routing protocols, power-aware routing protocols.

Transport Layer: Issues in designing – Transport layer classification, adhoc transport protocols. Security issues in adhoc networks: issues and challenges, network security attacks, secure routing protocols.

Learning outcomes: At the end of this unit, the students will be able to

- Learn different routing protocols at network layer
- Learn different routing protocols at transport layer and security issues.

UNIT IV: **12 Lectures**

Quality of Service: Introduction, issues and challenges in providing QoS in adhoc wireless networks, classification of QoS solutions.

Cross Layer Design: Need for cross layer design, cross layer optimization, parameter optimization techniques, cross layer cautionary perspective.

Learning outcomes: At the end of this unit, the students will be able to

- Learn different Quality Of Service solutions
- Learn the framework for MANETs

UNIT V: **8 Lectures**

Energy Management: Introduction, need for energy management in adhoc wireless networks, classification of energy management schemes, battery management schemes, transmission power management schemes, system power management schemes.

Learning outcomes: At the end of this unit, the students will be able to

- Learn different energy management schemes
- Recollect the power-aware routing protocols

Textbooks:

1. C. S. R. Murthy and B. S. Manoj, *Ad Hoc Wireless Networks: Architecture and Protocols*. New Delhi, India: Prentice Hall of India, 2004.
2. J. Sarangapani, *Wireless Ad Hoc and Sensor Networks: Protocols, Performance, and Control*. New Delhi, India: CRC Press, 2007.

References:

1. C. K. Toh, *Ad Hoc Mobile Wireless Networks: Protocols and Systems*, New Delhi, India: Pearson Education India, 2009.
2. C. S. Raghavendra and K. M. Sivalingam, *Wireless Sensor Networks*, USA: Springer Science, 2004.

Change of syllabus as compared to R-20 Regulations

There is 20% change in syllabus.

R20	R23
Transport Layer And Security Protocols: Introduction, issues in designing a transport layer protocol for adhoc wireless networks, design goals of a Transport Layer Protocol for Ad-Hoc Wireless Networks, classification of Transport Layer Solutions, TCP Over adhoc wireless networks, other Transport Layer Protocol for Ad-Hoc wireless Networks Security in adhoc wireless networks, network security requirements, issues and challenges in security provisioning.	Issues in designing – Transport layer classification, adhoc transport protocols. Security issues in adhoc networks: issues and challenges, network security attacks, secure routing protocols. **CROSS LAYER DESIGN:** Need for cross layer design, cross layer optimization, parameter optimization techniques, cross layer cautionary perspective.

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
MOBILE ADHOC NETWORKS	23IT5131(A)	**2025**	**Transport Layer:** Issues in designing – Transport layer classification, adhoc transport protocols. Security issues in adhoc networks: issues and challenges, network security attacks, secure routing protocolsCROSS LAYER DESIGN: Need for cross layer design, cross layer optimization, parameter optimization techniques, cross layer cautionary perspective.	**Entrepreneurship**

SOFT COMPUTING
(PROFESSIONAL ELECTIVE-3)

COURSE CODE: 23IT5131(B) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): Artificial Intelligence, DMS

Course Objectives:

The course would aim to make the student understand the basic idea of problem solving through the principles of soft computing, which would be seen as a well-balanced integration of fuzzy logic, evolutionary computing, and neural information processing.

- To introduce the ideas of fuzzy sets, fuzzy logic and use of heuristics based on human experience.
- To familiarize with genetic algorithms.
- To become familiar with neural networks that can learn from available examples and generalize to form appropriate rules for inferencing systems

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Recognize the feasibility of applying a soft computing methodology for a particular problem.
CO-2: Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems.
CO-3: Apply Hybrid systems for problem solving.
CO-4: Apply neural networks and genetic algorithms to pattern classification.

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2	2	3	2	1	1	1				1	3	2
CO-2	3	3	3	3	2	1	1	1				1	3	2
CO-3	3	3	3	3	3	1	1	1				1	3	2
CO-4	3	3	3	3	3	1	1	1				1	3	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **9 Lectures**

Overview of Soft Computing, Difference between Hard and Soft computing, Requirement of Soft computing, Major Areas of Soft Computing, Applications of Soft Computing

Learning outcomes: At the end of this unit, the students will be able to

- Understand the basics of fuzzy logic
- Illustrate the operations on fuzzy logic

UNIT II: **9 Lectures**

Neural Networks: Overview of Neural Networks, Learning rules and various activation functions, Single layer Perceptrons , Back Propagation networks, Architecture of Backpropagation Networks, Backpropagation, Variation of Standard Back propagation Neural Network, Introduction to Associative Memory, Adaptive Resonance theory and Self Organizing Map, Recent Applications.

L**earning outcomes**: At the end of this unit, the students will be able to

- Understand the architecture of Neural Networks
- Understand the framework of Back Propagation

UNIT III: **9 Lectures**

Fuzzy Systems: Fuzzy Set theory, Fuzzy versus Crisp set, Fuzzy Relation, Fuzzification, Minmax Composition, Defuzzification Method, Fuzzy Logic, Fuzzy Rule based systems, Predicate logic, Fuzzy Decision Making, Fuzzy Control Systems, Fuzzy Classification.

Learning outcomes: At the end of this unit, the students will be able to

- Explore different types of fuzzy models
- Apply fuzzy models to real world applications

UNIT IV: **9 Lectures**

Genetic Algorithms: Fundamentals of Genetic Algorithms: Basic Concepts Creation, Offspring's Encoding, Fitness functions, Reproduction, Genetic Modelling: Inheritance Operators, Cross over, Inversion and detection, Mutation operator, Bitwise operators, Multi-level Optimization.

Learning outcomes: At the end of this unit, the students will be able to

- Understand the importance of genetic algorithms
- Explore the applications of genetic algorithms

UNIT V: **9 Lectures**

Hybrid Systems: Sequential Hybrid Systems, Auxiliary Hybrid Systems, Embedded Hybrid Systems, Neuro-Fuzzy Hybrid Systems, Neuro-Genetic Hybrid Systems, Fuzzy-Genetic Hybrid Systems.

L**earning outcomes**: At the end of this unit, the students will be able to

- understand the Hybrid Systems
- Different Hybrid systems and their architecture

Textbooks:

1. S. Rajasekaran and G. A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications*. New Delhi, India: PHI Learning, 2003.
2. J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. New Delhi, India: PHI/Pearson Education, 2004.

References:

1. T. J. Ross, *Fuzzy Logic with Engineering Applications*. New York, NY, USA: McGraw-Hill (TMH), 1997.
2. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, 3rd ed. Boston, MA, USA: Addison-Wesley, 1989.
3. B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. New Delhi, India: Prentice Hall of India, 2006.
4. V. Kecman, *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*, 1st ed. Boston, MA, USA: Pearson Education, 2001.

Change of Syllabus:

Previous Unit No(R15)	Current Unit No (R19)	Changes Incorporated
UNIT IV	Unit-4 **Hybrid Systems**	**Newly introduced all topics**
UNIT V	Unit-2	Backpropagation Learning, Variation of Standard Back propagation Neural Network,
Overall change: 30%		

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/ Skill development/ Entrepreneurship
Soft Computing		R20	Hybrid Systems and Back Propagation Networks	Skill development

DIGITAL FORENSICS
(PROFESSIONAL ELECTIVE-3)

COURSE CODE: 23IT5131(C) CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): DBMS, Computer Networks, OS

Course Objective:

- To understand the basic digital forensics techniques for conducting the forensic examination on different digital devices.
- To understand how to examine digital evidences such as the data acquisition, identification analysis.

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Define the concept of ethical hacking and its associated applications in Information Communication Technology (ICT) world.

CO-2: Underline the need of digital forensic and role of digital evidences.

CO-3: Illustrate the methodology of incident response and various security issues in ICT world and identify digital forensic tool for data collection.

CO-4: List the method to generate legal evidence and supporting investigation reports and will also be able to use various digital forensic tools.

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	1		3	2	3					2	2	2
CO-2	3	2	1	1	3	2	3					2	2	3
CO-3	2	3	3	3	3	2	2	3				3	3	2
CO-4	2	3	3	2	2	2	2	3				3	2	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: Introduction to Forensics - Computer forensics **9 Lectures**

Fundamentals, Benefits of forensics, computer crimes, computer forensics evidence and courts, legal concerns and privacy issues, Introduction to Forensic life cycle and need

Learning outcomes: At the end of this unit, the students will be able to

- Understand the fundamentals of digital forensics.
- Identify legal procedures in relation to cyber-crime.

UNIT II: Investigations **8 Lectures**

Understanding Computing Investigations – Process of High-Tech investigations, Understanding data recovery work station and software, conducting investigations, Seriousness of the issues

Learning outcomes: At the end of this unit, the students will be able to

- Understand the procedures for conducting investigations.
- Conduct investigations in relation to cyber-crime.

UNIT III: Data Acquisition and its tools **10 Lectures**

Data acquisition - Understanding storage formats and digital evidence, determining the best acquisition method, Using Acquisition Tools, validating data acquisitions, performing RAID data acquisitions, Using remote network acquisition tools, Using other forensics acquisitions tools, Retention.

Network Forensics: Overview of Network Forensics, performing live acquisitions, developing standard procedures for network forensics, using network tools, examining the honeynet project – Introduction and Purpose, Darkweb

Case study on disk analysis tools - Autopsy/the Sleuth Kit, volatility

Learning outcomes: At the end of this unit, the students will be able to

- Understand the data acquisition tools
- Analyze several formats of data during a forensic case

UNIT IV: Case Analysis **10 Lectures**

Processing crimes and incident scenes, securing a computer incident or crime, seizing digital evidence at scene, storing digital evidence, obtaining digital hash, reviewing case.

E-mail investigations: Exploring the role of email in investigations, exploring the role of client and server in email, investigating email crimes and violations, understanding email servers, using specialized email forensic tools.

Case study on Digital forensic tool kit – FTK Imager

Learning outcomes: At the end of this unit, the students will be able to

- Idealize a crime scene.
- Analyze crime scene and collecting evidences.

UNIT V: Digital Forensics Tools **10 Lectures**

Current computer forensics tools - software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions, E-Mail investigations - investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool.

Learning outcomes: At the end of this unit, the students will be able to

- Understand various tools in relation to cyber forensics.
- Analyze several crime related to E-mail.

Textbooks:

1. W. G. Kruse II and J. G. Heiser, *Computer Forensics: Incident Response Essentials*. Boston, MA, USA: Addison-Wesley, 2002.
2. B. Nelson, A. Phillips, F. Enfinger, and C. Stuart, *Guide to Computer Forensics and Investigations*, 2nd ed. Boston, MA, USA: Thomson Course Technology, 2006.

Reference Books:

1. J. Vacca, *Computer Forensics: Computer Crime Scene Investigation*, 2nd ed. Hingham, MA, USA: Charles River Media, 2005.
2. B. Nelson, A. Phillips, F. Enfinger, and C. Steuart, *Guide to Computer Forensics and Investigations*, 2nd ed. Boston, MA, USA: Thomson Course Technology, 2006.
3. J. R. Vacca, *Computer Forensics: Computer Crime Investigation*. New Delhi, India: Firewall Media, 2009.
4. AccessData, *Forensic Toolkit (FTK) User Guide*, Version 7.4.2, February 2021. [Online]. Available: https://www.exterro.com/uploads/documents/FTK_7.4.2_UG.pdf d1kpmuwb7gvu1i.cloudfront.net+4

Change of Syllabus:

Unit No	Changes Incorporated
3	**Network Forensics:** Network forensic overview, performing live acquisitions, developing standard procedures for network forensics, using network tools, examining the honeynet project.
4	**E-mail investigations:** Exploring the role of email in investigations, exploring the role of client and server in email, investigating email crimes and violations, understanding email servers, using specialized email forensic tools.
Change of Syllabus: 10%	

Activity-Based Learning (Suggested Activities in Class)/ Practical Based learning:

- Flip Class
- Seminar/Poster Presentation
- Role play/Team Demonstration/Collaborative Activity
- Mini Project
- Case study
- Learn by Doing

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship
Digital Forensics	23IT5131	R23 AY: 2025-26	• Hands-on analysis of real-world cybercrime cases. • Tracing crypto transactions for fraud and cybercrime investigations. • Analyzing malware behavior and extracting evidence from compromised systems. • Hands-on experience with tools like **Autopsy, Wireshark, Volatility, Sleuth Kit**.	Employability/Skill Development

SOFTWARE TESTING METHODOLOGIES
(PROFESSIONAL ELECTIVE-3)

COURSE CODE: 23IT5131(D) CREDITS 3

L T P S Sessional Marks: 40

3 0 0 0 End Exam: 3 Hours

End Exam Marks: 60

Prerequisites: Software Engineering, JAVA

Course Objectives:

- To Understand the basics of software testing and explore different testing methods like unit, integration, regression, and system testing.
- To Learn how to plan a testing project, create test cases and data, perform tests, handle defects, and prepare test reports.
- To Get familiar with advanced testing concepts, including testing for web-based, and component-based software.
- To Understand the challenges of software test automation and how to solve them.
- To Develop skills to write test documents and effectively communicate with engineers in different formats.

Course Outcomes:

After course completion, the students are able to :

CO-1: Identify and understand various software testing problems, apply software testing knowledge and engineering methods and solve these problems by designing and selecting software test models, criteria, strategies, and methods.

CO-2: Design and implement a software testing process for a project, evaluate test automation requirements, and utilize diverse communication techniques to coordinate and execute practice-oriented software testing activities with team members.

CO-3: Basic understanding and knowledge of contemporary issues in software testing, such as component-based, web based testing problems.

CO-4: Write test cases for given software to test it before delivery to the customer and write test scripts for both desktop and web-based applications.

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	3	2	1	2	1	1	1	2	2	3	3	2
CO-2	3	3	3	3	1	3	2	2	3	3	3	3	3	3
CO-3	3	3	2	2	1	2	2	1	2	2	2	3	2	2
CO-4	3	3	3	3	3	2	1	1	2	3	2	3	3	3

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **8 Lectures**

Introduction to Software Testing: Introduction, Evolution, Myths & Facts, Goals, Psychology, definition, Model for testing, Effective vs Exhaustive Software Testing.

Software Testing Terminology and Methodology: Software Testing Terminology, Software Testing Life Cycle, Software Testing Methodology.

Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, High level and low-level designs, verifying code, Validation.

Learning Outcomes: At the end of this Unit the students will be able to

- Understand software testing concepts, evolution, and goals.
- Learn testing terminology, life cycle, and methodologies.
- Differentiate and apply verification & validation.

UNIT II: **8 Lectures**

Dynamic Testing - Black Box testing techniques: Boundary Value Analysis, Equivalence class Testing, State Table based testing, Decision table-based testing, Cause-Effect Graphing based testing, Error guessing.

White-Box Testing: Need, Logic Coverage criteria, Basis Path testing, Graph matrices, Loop testing, data flow testing, mutation testing.

Learning Outcomes: At the end of this Unit the students will be able to

- Apply black-box testing techniques effectively.
- Implement white-box testing methods like path and loop testing.
- Analyze logic coverage and data flow testing.

UNIT III: **8 Lectures**

Static Testing: Inspections, Structured Walkthroughs, Technical Reviews.

Validation Activities: Unit testing, Integration Testing, Function testing, system testing, acceptance testing.

Regression Testing: Progressives vs regressive testing, Regression test ability, Objectives of regression testing, Regression testing types, Regression testing techniques.

Learning Outcomes: At the end of this Unit the students will be able to

- Learn static testing techniques like reviews and inspections.
- Understand validation activities in software testing.
- Gain knowledge of regression testing types and techniques.

UNIT IV: **10 Lectures**

Efficient Test Suite Management: Growing nature of test suite, Minimizing the test suite and its benefits, test suite prioritization, Types of test case prioritization, prioritization techniques, measuring the effectiveness of a prioritized test suite .

Software Quality Management: SQ Metrics – Product Quality Metrics, In-Process Quality Metrics, Metrics for Software Maintenance, SQA models

Debugging: Process, Techniques, Correcting bugs.

Learning Outcomes: At the end of this Unit the students will be able to

- Manage test suites with minimization and prioritization.

- Understand software quality metrics and SQA models.
- Learn debugging techniques for effective defect correction.

UNIT V: **12 Lectures**

Automation and Testing Tools: Need for automation, categorization of testing tools, selection of testing tools, Cost incurred, Guidelines for automated testing, Overview of some open source testing tools - Jira, LOCUST, JUNIT, Test Automation using Selenium tool.

Testing Web based Systems: Challenges in testing for web-based software, quality aspects, web engineering, testing of web based systems, Testing mobile systems.

Learning Outcomes: At the end of this Unit the students will be able to

- Understand automation needs, tool selection, and costs.
- Gain knowledge of testing tools like Selenium, JMeter, and Taurus.
- Learn web and mobile application testing challenges.

CASE STUDY: Introduction to Selenium testing tool - Installation process, how to write and run a test case in Selenium.

Textbooks:

1. N. Chauhan, *Software Testing: Principles and Practices*. Oxford: Oxford University Press.
2. Y. Singh, *Software Testing*. Cambridge, U.K.: Cambridge University Press, 2011.

References:

1. P. C. Jorgensen, *Software Testing: A Craftsman's Approach*, 3rd ed. Boston, MA, USA: Auerbach Publications (Distributed by SPD), 2008.
2. P. Ammann and J. Offutt, *Introduction to Software Testing*. Cambridge, U.K.: Cambridge University Press, 2008.
3. P. Nageswara Rao, *Software Testing: Concepts and Tools*. New Delhi, India: Dreamtech Press, 2007.
4. N. Chauhan, *Software Testing: Principles and Practices*. New Delhi, India: Oxford University Press, 2010.

Change of Syllabus:

- 20% of syllabus has changed. Introduced in R20 regulation.

	Unit wise changes made
Unit 1	Verification and Validation concept is added.
Unit 3	Unit 3 has been partitioned into unit 2 (Blackbox and Whitebox testing) and unit 3(Static testing and Regression Testing).
Unit 4	Software Quality Management concept is added.
Unit 5	New concepts were introduced i.e. Automation and Testing tools & Testing Web based Systems.

1.1.3 of NAAC

Name of the Course	Couse Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/ Skill development/ Entrepreneurship
Software Testing Methodologies	23IT5131	2023	Black Box testing techniques, White-Box Testing techniques, Static and Regression Testing's	Skill Development

DESIGN AND ANALYSIS OF ALGORITHMS

COURSE CODE: 23IT4121 CREDITS 3
L T P S Sessional Marks: 40
3 0 0 0 End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): Introduction to programming, Data structures.

Course Objectives:

- Make students understand how asymptotic notations are used to provide a rough classification of algorithms.
- Explain different computational models (e.g., divide-and-conquer), complexity measures (e.g., running time) to analyze the complexity/performance of different algorithms.
- Explain various advanced design and analysis techniques such as greedy algorithms, dynamic programming & know the concepts of tractable and intractable problems.

Course Outcomes:

After completion of this course, students will be able to:

CO-1: Evaluate time complexities of various brute-force and other algorithms.
CO-2: Design algorithms to solve problems using divide and conquer, decrease and conquer, Dynamic programming and Greedy Techniques.
CO-3: Transform and solve the problems with known algorithms.
CO-4: Solve the problems using tractable algorithms (Backtracking and Branch-and-Bound).

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	2	3								1	2	2
CO-2	3	3	3	2	2						2	3	3	
CO-3	3	3	1									2	2	
CO-4	3	3	1	2								1	2	

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: **12 Lectures**

Introduction: Fundamentals of algorithmic problem solving.

Fundamentals of analysis of algorithms and efficiency: Analysis framework –Asymptotic Notations and Basic Efficiency classes –Mathematical Analysis of Non-recursive Algorithms –Mathematical Analysis of recursive Algorithms.

Brute Force –Selection Sort, Bubble sort, Sequential Search, Brute Force String Matching, Exhaustive search problems, Closest-Pair and Convex-Hull Problems by brute Force

Learning outcomes: At the end of this unit the students will be able to

- Understand the concepts of time complexity, worst case, average case and best case complexities.
- Evaluate the time complexities of various algorithms under Brute force technique.

UNIT II: **12 Lectures**

Divide-and-Conquer: Merge sort, Quicksort, Binary Search, Strassen's Matrix Multiplication, Closest-Pair and Convex-Hull Problems by Divide-and-Conquer.

Decrease –and –Conquer: Decrease by a Constant Algorithms -Insertion Sort, Depth First Search and Breadth First Search, Topological Sorting, Decrease by a Constant Factor and Variable Size Decrease Algorithms.

Learning outcomes: At the end of this unit the students will be able to

- Understand Divide-and-Conquer and Decrease –and –Conquer strategies to solve problems.
- Evaluate time complexities of various algorithms under Divide-and-Conquer and Decrease –and –Conquer techniques.

UNIT III: **10 Lectures**

Transform-and-Conquer: Presorting, Balanced Search Trees (AVL), Heaps and Heapsort, Problem Reduction.

Space and Time Tradeoffs – Sorting by Counting, Input Enhancement in string Matching – Horspool's algorithm, B-Trees.

Learning outcomes: At the end of this unit the students will be able to

- Understand a wide range of searching and sorting algorithms.
- Evaluate time complexities of various algorithms under the Transform-and-Conquer technique.

UNIT IV: **10 Lectures**

Dynamic Programming: Warshall's and Floyd's Algorithms, Optimal Binary Search Trees, The Knapsack Problem-0/1 Knapsack Problem

Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees, **The Knapsack Problem-0/1 Knapsack Problem.**

Learning outcomes: At the end of this unit the students will be able to

- Understand Dynamic Programming and Greedy Techniques to solve problems.
- Evaluate time complexities of various algorithms under Dynamic Programming and Greedy Technique.

UNIT V: **10 Lectures**

Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP, NP hard and NP complete problems.

Coping with the Limitations of Algorithms Power: Backtracking - n queens, Hamiltonian circuit, subset sum problem.

Branch and Bound - Assignment Problem, knapsack Problem, Traveling salesman problem.

Learning outcomes:

At the end of this unit the students will be able to

- Understand the notion of tractable and intractable problems.
- Understand the notion of P, NP and NP-complete class problems.

Textbooks:

1. A. Levitin, *Introduction to the Design and Analysis of Algorithms*, 2nd ed. Boston, MA, USA: Pearson, 2009.
2. E. Horowitz, S. Sahni, and S. Rajasekaran, *Computer Algorithms in C++*, 2nd ed. Hyderabad, India: Universities Press, 2014.

References:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. New Delhi, India: PHI Learning, 2009.
2. S. Sridhar, *Design and Analysis of Algorithms*. New Delhi, India: Oxford University Press, 2013.

Change of Syllabus: No Change in R23 Regulation same as R20 Regulation

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/ Skill development/ Entrepreneurship**
Design and Analysis of Algorithms	23IT4121	R19	Mathematical Analysis of Non-recursive Algorithms, Mathematical Analysis of recursive Algorithms, Merge sort, Quicksort, Binary Search, Decrease by a Constant Algorithms, Balanced Search Trees (AVL), Heaps and Heapsort, Optimal Binary Search	Employability

APPLIED CRYPTOGRAPHY

COURSE CODE: 23IT4122
L T P S
3 0 0 0

CREDITS 3
Sessional Marks: 40
End Exam: 3 Hours
End Exam Marks: 60

Prerequisite(s): DMS, DLD, CO

Course Objectives:

- Understand the fundamental concepts of **number theory** and their applications in cryptographic algorithms.
- Learn and implement **symmetric and asymmetric encryption techniques**, including block ciphers, stream ciphers, and public-key cryptography.
- Analyze and apply **hash functions and message authentication codes (MACs)** for ensuring data integrity and authenticity.
- Explore **real-world cryptographic applications**, including side-channel analysis, blockchain technology, and key establishment protocols.

Course Outcomes

CO-1: Apply number theory concepts such as modular arithmetic, GCD, and Chinese Remainder Theorem in cryptographic algorithms.
CO-2: Implement and analyze symmetric encryption techniques including Block and Stream Ciphers with different encryption modes.
CO-3: Demonstrate the working of Public Key Cryptography, Hash Functions, and Message Authentication Codes for data security.
CO-4: Explore advanced cryptographic techniques such as Side Channel Analysis and real-world applications like Blockchain and Key Establishment Protocols.

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	3	2	2	1	-	-	-	-	-	-	2	3	3
CO-2	3	3	3	3	2	-	-	-	-	-	2	3	3	3
CO-3	3	3	3	3	2	-	-	-	-	-	2	3	3	3
CO-4	3	3	3	3	2	-	-	-	-	-	2	3	3	3

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

UNIT I: Foundations of Cryptography **9 Lectures**

Terminology, Substitution Ciphers - **Monoalphabetic** ciphers (e.g., Caesar, Simple Substitution), **Polyalphabetic** ciphers (e.g., Vigenère, Autokey), **Polygraphic** ciphers (e.g., Playfair, Hill), Transposition Ciphers, One-Time Pads

Cryptanalysis, Real-world failures – WEP protocol weakness, Improper Key reuse, Birthday attack in hash collisions

Quantum Cryptography - Introduction

Learning outcomes: At the end of this unit the students will be able to

- Understand and apply classical cryptographic techniques such as substitution and transposition ciphers, including monoalphabetic, polyalphabetic, and polygraphic systems, to encrypt and decrypt messages.
- Analyze cryptographic vulnerabilities through the study of cryptanalysis methods, real-world failures (e.g., WEP protocol, key reuse issues, and birthday attacks), and gain a conceptual understanding of emerging paradigms like quantum cryptography.

UNIT II: Symmetric Cryptography **9 Lectures**

Techniques and principles behind cryptography

Block Ciphers: ECB, CBC, OFB, CFB, CTR and GCM modes, AES, Twofish

Stream Ciphers: Encryption and decryption with Stream ciphers, Shift-register based stream ciphers, (ChaCha, RC4, Salsa20).

Learning outcomes: At the end of this unit the students will be able to

- Explain the core principles and techniques of symmetric key cryptography, and distinguish between block and stream ciphers based on their design and operational characteristics.
- Apply and analyze various symmetric encryption modes (e.g., ECB, CBC, OFB, CFB, CTR, GCM) and cipher algorithms (e.g., AES, Twofish, ChaCha, RC4, Salsa20) for secure communication.

UNIT III: Public Key Cryptography **9 Lectures**

RSA, ElGamal, Diffe-Hellman Key exchange, practical digital signatures, ECC, Digital Signatures, Certificate Authorities.

Learning outcomes: At the end of this unit the students will be able to

- Understand and apply public key cryptographic algorithms such as RSA, ElGamal, Diffie-Hellman, and Elliptic Curve Cryptography (ECC) for secure key exchange and encryption.
- Explain the role of digital signatures and certificate authorities in ensuring data authenticity, integrity, and trust in digital communication systems.

UNIT IV: Cryptographic Hash Functions & Authentication **9 Lectures**

Hash Functions: One-way, collision resistant, preimage resistant HASH functions – SHA512, Real-world examples.

Message Authentication Codes: MAC from Hash functions, MAC from block ciphers.

Password Security: Key Derivation Functions (PBKDF2, Argon2, bcrypt, scrypt).

Learning outcomes: At the end of this unit the students will be able to

- Explain the properties and applications of cryptographic hash functions (e.g., SHA-512), and evaluate their role in ensuring data integrity and resistance to attacks such as collisions and preimage attacks.

- Analyze and implement authentication mechanisms using Message Authentication Codes (MACs) and key derivation functions (e.g., PBKDF2, Argon2, bcrypt, scrypt) for secure password storage and verification.

UNIT V: Cryptographic Attacks & Applications **9 Lectures**

Side-Channel Attacks: Power analysis, timing attacks, fault injection.

Secure Protocols & Applications: Key Establishment Protocols, Blockchain, Zero-Knowledge Proofs, Secure Multi-Party Computation.

Modern Cryptographic Techniques: Homomorphic Encryption, Functional Encryption, Secure Boot, TPMs.

Learning outcomes: At the end of this unit the students will be able to

- Analyze various side-channel attacks such as power analysis, timing attacks, and fault injection, and evaluate appropriate countermeasures to mitigate these threats.
- Apply modern cryptographic techniques like homomorphic encryption, zero-knowledge proofs, and secure multi-party computation in designing secure protocols and applications, including blockchain and trusted hardware (TPMs, Secure Boot).

Textbooks:

1. B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed. New York, NY, USA: Wiley, 1996.
2. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL, USA: CRC Press, 1996.

References:

1. W. Stallings, *Cryptography and Network Security: Principles and Practices*, 6th ed. New Delhi, India: Pearson Education, 2014.
2. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, 5th printing. Boca Raton, FL, USA: CRC Press, 2001. [Online]. Available: https://www.crcpress.com
3. D. Mukhopadhyay and R. S. Chakraborty, *Hardware Security: Design, Threats, and Safeguards*. Boca Raton, FL, USA: CRC Press, Taylor and Francis Group, 2019. [Online]. Available: https://www.crcpress.com

Change of syllabus as compared to R-20 Regulations

Unit No	R20 Regulations	R23 Regulations	Changes Incorporated
Unit 1	Cryptographic fundamentals, Number Theory, Classical Ciphers	Real-world failures – WEP protocol weakness, Improper Key reuse, Birthday attack in hash collisions	Real-world failures – WEP protocol weakness, Improper Key reuse, Birthday attack in hash collisions
Unit 2	Symmetric Encryption: DES, AES, Feistel Networks	Symmetric Cryptography: Block Ciphers (AES, Twofish), Stream Ciphers (ChaCha, RC4,	Expanded content on stream ciphers and encryption modes

		Salsa20), Key Management	
Unit 3	Public Key Cryptography: RSA, ElGamal, ECC	Public Key Cryptography: RSA, ElGamal, Diffie-Hellman, Digital Signatures, ECC	ECC
Unit 4	Hash Functions, Message Authentication Codes (MACs)	Cryptographic Hash Functions & Authentication: Hash Functions, MACs, Password Security (PBKDF2, Argon2, bcrypt, scrypt)	More focus on password security and key derivation functions
Unit 5	Network Security & Cryptographic Applications	Cryptographic Attacks & Applications: Side-Channel Attacks, Secure Protocols, Blockchain, Zero-Knowledge Proofs, Secure Multi-Party Computation	Added side-channel attacks, blockchain, and zero-knowledge proofs
	Overall change: 40%		

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
Applied Cryptography	23IT4122	R23 AY: 2025-26	• Implement various cryptographic algorithms (AES, RSA, ECC) • Enhances Analytical thinking of various cryptographic protocols (TLS/SSL, Blockchain Security) • Develop expertise in cybersecurity and secure communications. • Essential for Real-world security applications and attack mitigation	Employability

CRYPTOGRAPHY LAB

COURSE CODE: 23IT4221
L T P S
0 0 3 0

CREDITS 1.5
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s): DMS

Course Objectives:

- Understand the principles and practices of cryptographic techniques.
- Understand information security goals for designing secure systems.
- Apply security algorithms in solving real-life security problems in communicating systems.
- Apply security to information over the network and world wide web.

Course Outcomes:

After completion of this course, the students will be able to

CO-1: Implement and Analyze Classical and Modern Cipher Techniques
CO-2: Develop and Apply Secure Cryptographic Protocols
CO-3: Evaluate Cryptographic Security and Vulnerabilities

Mapping of Course Outcomes with Pos and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	2	3	3	1	-	-	-	-	1	-	1	2	2	-
CO-2	2	3	3	3	3	1	-	-	2	-	2	3	2	1
CO-3	-	2	-	2	3	1	1	-	3	2	3	3	-	-

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

List of Experiments

1. Implement a monoalphabetic substitution cipher to encrypt and decrypt text using a fixed shift. – CO-1
2. Implement a polyalphabetic substitution cipher using a keyword for encryption and decryption. – CO-1
3. Implement a transposition cipher to rearrange plaintext characters based on a rail pattern. – CO-1
4. Simulate a simple block cipher in ECB mode using XOR operations for encryption and decryption. – CO-1
5. Implement a basic stream cipher using a repeating key stream for XOR-based encryption and decryption. – CO-1
6. Implement a simplified RSA algorithm for encryption and decryption using small prime numbers. – CO-2
7. Create a basic hash function that computes a hash value by summing ASCII values modulo a constant. – CO-3
8. Implement password hashing and verification using PBKDF2 with a random salt. – CO-2
9. Simulate a timing attack by comparing strings with intentional delays to demonstrate side-channel vulnerabilities. – CO-3

10. Implement a simple zero-knowledge proof protocol (e.g., graph coloring) to demonstrate authentication without revealing secrets. – CO-2

Textbooks:

1. W. Stallings, *Cryptography and Network Security: Principles and Practice*, 7th ed. Boston, MA, USA: Pearson Education, 2017.
2. B. A. Forouzan, *Cryptography and Network Security*, 3rd ed. New York, NY, USA: McGraw Hill, 2015.

References:

1. B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed. New York, NY, USA: Wiley, 1996.
2. A. Kahate, *Cryptography and Network Security*, 3rd ed. New York, NY, USA: McGraw Hill, 2019.

Change of syllabus as compared to R-20 Regulations:

- **Newly Introduced Lab**

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/Skill development/ Entrepreneurship**
Cryptography LAB	23IT4221	R23 AY: 2025-26	• Implementation of encryption and decryption techniques using classical and modern cryptographic algorithms. • Secure key exchange methods and analysis of security vulnerabilities. • Implementation of RSA for secure communication. • Hands-on experience with cryptographic tools like CrypTool.	Skill Development (Hands-on experience in cryptography implementation)

ETL TOOLS
(ELECTIVE LAB – 1)

COURSE CODE: 23IT5211(A) | CREDITS 1.5
L T P S | Sessional Marks: 50
0 0 3 0 | End Exam: 3 Hours
End Exam Marks: 50

Prerequisites:

- Basics of Data Warehousing
- Programming Knowledge preferably Python
- Database Fundamentals
- Basic understanding of data formats

Course Objectives:

- To explore different types of data and learn the key aspects of warehousing in extracting, transforming, and loading the warehouse with the data.
- To be able to perform the tasks using familiar tools such as Talend Open Studio, Apache Airflow

Course Outcomes:

At the end of the course, the student will be able to:

CO-1: Understand and implement the process of data extraction from various sources using ETL tools like Talend Open Studio, ensuring data is properly read, formatted, and integrated.
CO-2: Develop expertise in data transformation techniques such as filtering, sorting, joining, and mapping, while ensuring data quality and consistency for effective analysis.
CO-3: Demonstrate proficiency in loading transformed data into target storage systems, managing job execution, and orchestrating ETL processes for automation and efficiency.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	2	1	2	3	-	-	-	-	-	-	-	-	1	2
CO-2	2	3	1	3	-	-	-	-	-	-	-	-	1	2
CO-3	2	3	1	3	-	-	-	-	-	-	-	-	1	2

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Memory Requirements

- Memory usage: 4 GB minimum, 8 GB recommended
- Disk space: 20 GB

Software Requirements

- Java 8 JRE Oracle.
- A properly installed and configured MySQL database, with a databasenamed gettingstarted.
- Talend Open Studio

Course Outline

1) Data Extraction from Sources (CO-1)

- Identifying different types of data and sources.
- Extracting raw data from an array of sources including databases, network appliances, security hardware and software applications, and others.
- Identify relevant data necessary for transformation.

2) Data Profiling & Data Quality (CO-1)

- **Structure discovery** — Structure discovery (or analysis) helps determine whether your data is consistent and formatted correctly. It uses basic statistics to provide information about the validity of data.
- **Relationship discovery** — Relationship discovery identifies connections between different data sets.
- **Content discovery** — Content discovery focuses on data quality. Data needs to be formatted, standardized, and properly integrated with existing data in a timely and efficient manner. For example, if a street address is incorrectly formatted it could mean that certain customers can't be reached, or a delivery becomes misplaced.
- **Quality Analysis**
    - Connecting to a data source including databases and delimited file
    - Database content analysis
    - Column analysis
    - Table analysis
    - Redundancy analysis
    - Correlation analysis
    - Patterns and Indicators

3) Data Transformations (CO-2)

- Data cleaning
- Data recovery – using data profiling.
- Data mapping
- Generating code
- Executing the code
- Review
- Customized operations – Additional steps
    - Filtering (e.g., Selecting only certain columns to load).
    - Enriching (e.g., Full name to First Name, Middle Name, Last Name).
    - Splitting a column into multiple columns and vice versa.
    - Joining together data from multiple sources.
    - Removing duplicate data.

4) Data Loading to Target (CO-3)

The transformed, high quality data is then delivered to a single, unified target location for storage and analysis.

5) **Data Reconciliation (CO-3)**

- Setting up a reconciliation strategy for deployment conflicts
- Setting up the reconciliation strategy for deployment conflicts in preferences

Note: This course is related to Employability/Skill development.

List of Experiments

Week-1: Discovering Talend Open Studio (CO-1)

- Introduction to Talend Open Studio
- Understanding the process of installing the software
- Create a job to display the message "Welcome message"

Week-2: Working with Projects, Extracting Data (CO-1)

- Create a job to illustrate the process of Reading a delimited file and displaying it on the console

Week-3 (CO-1)

- Create a job to illustrate the process of Reading an excel file and displaying it on the console and redirecting it to an external source
- Create a job to illustrate the process of Reading an arff file and displaying it on the console and redirecting it to an external source
- Create a job to illustrate the process of working on tFileInputFullRow
- Create a job to illustrate the process of working on tFileInputMail

Week-4: Managing Files (CO-2)

- Create a job to illustrate the process of Working on tFileList
- Create a job to illustrate the process of Working on tFileCopy
    - Copying and removing files
    - Renaming files
    - Time stamping files
- Create a job to illustrate the process of Working on tFileExist and deleting the file using tFileDelete
- Create a job to illustrate the process of Working on tFileArchive / tFileUnarchive

Week-5: Filtering, Sorting, and other processing techniques (CO-2)

- Create a job to group the data using **tAggregateRow**(Group By) component
- Create a job to group the data and display it in sorted order using **tAggregatedSortedRow** component
- Create a job to illustrate the concept of **tFilterColumns**
- Create a job to illustrate the concept of **tFilterRow**
- Create a job to illustrate the concept of **tSortRow**
- Create a job to demonstrate data normalization
- Create a job to demonstrate data denormalization
- Create a job to illustrate the concept of **tJoin**
- Create a job to illustrate the concept of **tMap**

Week-6: Transformation (CO-2)

- Transform XML to CSV
- Transforming CSV to XML
- Maps and expressions
    - CSV file is a customer datafile and has the following fields:
        - Customer ID
        - First Name
        - Last Name

- Address1
- Address2
- Town City
- County
- Postcode
- Telephone

The XML file we want to produce has a similar, yet different set of fields, as follows:

XML Field	CSV Mapping
id	Customer ID field (left-padded with zeros to make its length 8)
name	First Name and Last Name
address_1	Address1 and Address2
address_2	Town City, County, and Postcode
telephone_number	Telephone with non-numeric characters removed

- Create a job that extracts data from multiple sheets in a excel file.

Week-7: Working with Databases (CO-2)

- Create a job to illustrate the process of connecting to a database (Oracle / MySql)
- Create a job that illustrates the process of extracting data from a database table (Oracle/MySql)
- Create a job that illustrates the process of extracting data from multiple tables in a database (Oracle/MySql)
- Create a job to illustrate the process of writing data in to a database file (Oracle/MySql)
- Create a job to illustrate the process of modifying data in a database file (Oracle/MySql)
- Create a job to demonstrate database to database transfer
- Demonstrate the usage of SQL templates on considering a usecase.
  Create a Job that:
  - opens a connection to a Mysql database.
  - collects data grouped by specific value(s) from a database table and writes aggregated data in a target database table.
  - deletes the source table where the aggregated data comes from.
  - reads the target database table and lists the Job execution result.

Week-8: Job Orchestration (CO-3)

- Breaking the overall integration job into discrete tasks using subjobs
- Defining flow logic within our jobs so that certain tasks are invoked only if specific conditions apply
- Providing checkpoints where we can trap errors or failures
- Using iterate and loop concepts to cycle through a number of executions of a specific task
- Splitting and merging dataflows

Week-9: Managing Jobs (CO-3)

- Code versioning and how we can use the Studio to manage iterations of the same job
- Exporting and importing jobs from the Studio for collaboration purposes or to back up
- Exporting jobs for standalone execution
- Scheduling jobs for automated execution

Week-10: Global Variables and contexts (CO-3)

- **Studio global variables**: The variables that the Studio makes available through the components we use in our integration jobs
- **User defined global variables**: Ad-hoc variables that can be configured in your jobs
- **Job contexts**: The variables we can create to execute jobs with different parameters for different environments or scenarios

Textbooks:

1. J. Bowen, *Getting Started with Talend Open Studio for Data Integration*. Birmingham, U.K.: Packt Publishing, 2012.

References:

1. R. Barton, *Talend Open Studio Cookbook*. Birmingham, U.K.: Packt Publishing, 2013. ISBN 978-1-78216-726-6.
2. *Talend Open Studio for Data Quality User Guide*. [Online]. Available: https://www.talend.com. Creative Commons Public License.
3. *Talend Open Studio for Big Data User Guide*, 2017. [Online]. Available: https://www.talend.com. Creative Commons Public License.
4. R. Kimball and J. Caserta, *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data*. Hoboken, NJ, USA: Wiley Publishing, 2004.

Change of Syllabus

Category	Old Syllabus (R19)	New Syllabus (R23)	Change (%)
CO-1	Consolidate data from different sources	Understand and implement data extraction	15% (Shifted focus to Extraction)
CO-2	Refine data during transformation	Develop expertise in transformation techniques	5% (Refined wording)
CO-3	Load transformed data for storage and analysis	Manage job execution and automation	10% (Expanded to include orchestration)
Week-1	Discovering Talend Open Studio	Same	0%
Week-2	Working with Projects, Extracting Data	Same	0%
Week-3	Quality Analysis	Added Reading Excel, ARFF, File Input Processing	20% (Expanded to include additional file formats)
Week-4	Transformation	Managing Files (File Copy, Delete, Archive, etc.)	Rearranged
Week-5	Working with Databases	Filtering, Sorting, Joins, Data Mapping	Rearranged
Week-6	Filtering, Sorting, Joins, Data Mapping	Data Transformation (XML-CSV, Mapping, Data Extraction)	Rearranged
Week-7	Managing Files	Working with Databases (SQL, MySQL, Oracle,	Rearranged

Category	Old Syllabus (R19)	New Syllabus (R23)	Change (%)
		Extraction)	
Week-8	Job Orchestration	Same	0%
Week-9	Managing Jobs	Same	0%
Week-10	Global Variables & Contexts	Same	0%
Week-11	Loading (Publishing)	Merged into Data Loading in Week 4	10% (Merged)
Week-12	Auditing and Reconciliation	Merged into Data Reconciliation in Week 5	10% (Merged)
Week-13	Archive and Cleanup	Removed	10% (Removed but partially merged)

Key Changes in the New Syllabus (R23)

1. **Course Outcomes Revised:**
   - **CO-1 shifted focus** from "Consolidating" data to "Extracting" data.
   - **CO-3 expanded** to include automation and job execution.
2. **Rearranged Topics:**
   - **Managing Files moved from Week-7 to Week-4.**
   - **Filtering & Sorting moved from Week-6 to Week-5.**
   - **Working with Databases moved from Week-5 to Week-7.**
3. **Merged & Removed Topics:**
   - **Loading (Publishing) merged** into Week-4 (Data Loading).
   - **Auditing & Reconciliation merged** into Week-5 (Data Quality).
   - **Archive & Cleanup removed** (partially merged into other topics).
4. **Expanded Practical Coverage:**
   - **Week-3 now includes Excel, ARFF, and File Processing.**
   - **More focus on extracting and handling structured/unstructured data.**

Overall Change Percentage: ~30-35%

- **Expanded Content: 15%**
- **Rearranged Topics: 10%**
- **Merged/Removed Topics: 5-10%**

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/Entrepreneurship
ETL Tools	23IT5211(A)	R23	Working with Projects, Extracting Data Quality Analysis Transformation Working with Databases Filtering, Sorting, and other processing techniques Managing Files Auditing and Reconciliation	Skill Development

DEVOPS
(ELECTIVE LAB – 1)

COURSE CODE: 23IT5211(B) CREDITS 1.5
L T P S Sessional Marks: 50
0 0 3 0 End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s): Operating System, Virtualization, Cloud Computing, Java and Web Programming, and Software Engineering & Monitor the Software Applications

Course Objectives:

DevOps improves collaboration and productivity by automating infrastructure and work flows and continuously measuring applications performance.

Course Outcomes:

After completion of this course, the students will be able to

CO-1: Remember the importance of DevOps tools used in software development life cycle

CO-2: Analyze & Illustrate the Containerization of OS images and deployment of applications over Docker.

CO-3: Summarize and synthesize the importance of Software Configuration Management in DevOps using Chef/Puppet/Ansible or Saltstack

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	1	1						1		1		1	1	1
CO-2	1	2	2	2	3			1	1	1	2	1	1	1
CO-3	1	2	2	2	3			1	1	1	2	1	1	1

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Hardware & Software Requirements:

PC with following Configuration

1. Windows or Linux Desktop OS for Client machines
2. Internet Connection for each PC with at least 2 MBPS
3. Intel Core i3/i5/i7 Processor with Intel VT-X support
4. 4 GB RAM Minimum
5. 500 GB Hard Disk
6. Gigabit Ethernet (GbE) network interface card (NIC CentOS / Fedora / Ubuntu / Redhat Server OS for One Server)
7. JDK or higher
8. Netbeans or Eclipse
9. Open SSH.
10. List of Software‘s- Maven, Jenkins

S. No	Module / Experiment Number	Detailed Content	CO Mapping
1	Introduction	To Understand the Concept of DevOps with related technologies which are used to Code, Build, Test, Configure & Monitor The Software Applications	CO-1
2	Version Control	To perform Version Control on websites/Softwares using different Version control tools like RCS/ CVS/GIT (Any one)	CO-1
3	Virtualization & Containerization	To Install and Configure Docker for creating Containers of different Operating System Images.	CO-1 CO-2
4	Virtualization & Containerization	To Build, Deploy and manage web or Java Application on Docker	CO-1 CO-2
5	Virtualization & Containerization	To Deploy Microservices using Docker	CO-1 CO-2
6	Software Configuration Management	To install and configure Software Configuration Management using Chef / Puppet / Ansible or Saltstack.	CO-1 CO-3
7	Build & Test Applications with Continuous Integration	To Install and Configure Jenkins to test, and deploy Java or Web Applications using NetBeans or eclipse, maven	CO-1 CO-2

Textbooks:

1. S. Chacon and B. Straub, *Pro Git*, 2nd ed. Berkeley, CA, USA: Apress, 2014.
2. K. Matthias and S. P. Kane, *Docker: Up and Running*. Sebastopol, CA, USA: O'Reilly Media, 2015.
3. L. Bass, I. Weber, and L. Zhu, *DevOps: A Software Architect's Perspective*. Boston, MA, USA: Addison-Wesley, 2015.
4. J. F. Smart, *The Definitive Guide to Jenkins*. Sebastopol, CA, USA: O'Reilly Media, 2011.

Reference Books:

1. S. Sharma and B. Coyne, *DevOps for Dummies*. Hoboken, NJ, USA: Wiley, 2017.
2. M. Httermann, *DevOps for Developers*. Berkeley, CA, USA: Apress, 2017.
3. J. Verona, *Practical DevOps*. Birmingham, U.K.: Packt Publishing, 2017.

Change of Syllabus

There is 10% change in R23 syllabus.

1. Experiment 5 is added- Virtualization & Containerization.
2. Re-structured the order of experiments.

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/Skill development/ Entrepreneurship
DevOps	23IT5211	2021-22	Experiments 3, 4, 5 and 6. 1. Version Control 2. Virtualization & Containerization 3. Virtualization & Containerization 4. Software Configuration Management	Skill Development
DevOps	23IT5211	2025-26	Experiment 5 is added. Virtualization & Containerization	Skill Development

ADVANCED JAVA
(ELECTIVE LAB – 1)

COURSE CODE: 23IT5211(C) CREDITS 1.5
L T P S Sessional Marks: 50
0 0 3 0 End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s): Basic Knowledge on core Java Concepts, HTML tags and DBMS.

Course Objectives:

- Familiarize with User Interface (GUI), networking, and database manipulation.
- Understand the enterprise application concepts and HTTP protocol.
- Illustrate the concepts of Java Servlets and Java Server Pages to develop web applications.

Course Outcomes:

After completion of this course student will be able to:

CO-1: Design and develop web applications using Servlets, JSP, JDBC, ORM, Hibernate, and J2EE integration. Utilize Spring MVC and Spring Boot frameworks for advanced web application development.
CO-2: Design and implement components like: Session, JSTL, Tag Extension and Filter.
CO-3: Distinguish Web Server, Web Container and Application Server.

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2	2	2	3				3	1	2	3	3	
CO-2	3	2	2	2	3				3	1	2	3	3	
CO-3	3	1			3							1	2	

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Course Outline

- J2EE and Web Development Java Platform, J2EE Architecture Types, Explore Java EE Containers, Types of Servers in J2EE Application, HTTP Protocols and API, Request Processing in Web Application, Web Application Structure, Web Containers and Web Architecture Models.
- Advance Networking Basics, Introduction of Socket, Types of Socket, Socket API, TCP/IP client sockets, URL,TCP/IP server sockets, Datagrams, java.net package, Server Socket, Client Server programming.
- JDBC Programming JDBC Architecture, Types of JDBC Drivers, Introduction to major JDBC Classes and Interface, Creating simple JDBC Application, Types of Statement (Statement Interface, Prepared Statement, CallableStatement), Exploring Result Set Operations. Connection Pooling and Performance Optimization Handling Large Objects (BLOB and CLOB) in JDBC, Integration of JDBC with ORM Framework. CRUD Operation using Hibernate API.
- Servlet API and Overview Servlet Introduction, Servlet Life Cycle, Types of Servlet, Servlet Configuration with Deployment Descriptor, Working with Servlet Context and Servlet Config Object, Attributes in Servlet, Response and Redirection using Request

Dispatcher and using send Redirect Method, Filter API, Manipulating Responses using Filter API, Session Tracking: using Cookies, HTTP Session.

- JSP architecture, JSP page lifecycle, JSP elements, Expression Language, Tag Extensions, Tag Extension API, Tag handlers, JSP Fragments, Tag Files, JSTL, Core Tag library, overview of XML Tag library. JSP action elements, MVC Architecture with JSP, Servlets, and Spring MVC, CRUD operations with JSP. Introduction to Spring Boot and J2EE Integration.

Textbooks:

1. *Black Book—Java Server Programming J2EE*, 1st ed. New Delhi, India: Dream Tech Publishers, 2008.
2. J. Keogh, *Complete Reference J2EE*. New York, NY, USA: McGraw-Hill, 2004.
3. S. Allamaraju and C. Buest, *Professional Java Server Programming*. Hoboken, NJ, USA: Wiley, 2005.
4. P. Sunderaraman, *BlackBook: Spring 3.0*. New Delhi, India: Kogent Solutions, Dream Tech Publications, 2010.

References:

1. *Spring Tutorial*, TPointTech. [Online]. Available: https://www.tpointtech.com/spring-tutorial. [Accessed: Apr. 15, 2025].
2. *Hibernate Tutorial*, TPointTech. [Online]. Available: https://www.tpointtech.com/hibernate-tutorial. [Accessed: Apr. 15, 2025].
3. *Spring Boot*, GeeksforGeeks. [Online]. Available: https://www.geeksforgeeks.org/spring-boot/. [Accessed: Apr. 15, 2025].

List of Programs:

SNo	PROGRAM	co
1.	**Week-1: Working with Net Beans:** Installing Net Beans and Oracle Database on Windows / Linux operating System.	3
2.	**Week-2: Socket Programming using Java.net package.** Develop a chatting application by establishing connection between client and server.	1
3.	**Week-3: JDBC Programs using Statement and Prepared Statement** 1)A program to test the connection with the database. 2)A program to create a table 3)A program to insert record in a table 4)A program to fetch records from a table 5)A program to update record in a table 6)A program to delete record from a table	1
4.	**Week-4: JDBC Programs using Callable Statement** 1) Create a store procedure which will insert one record in to employee table. 2) Create a store procedure which will retrieve salary for given employeeid. 3) Write a java application which will Call the above procedure and display appropriate information on screen.	1
5.	**Week-5 Perform CRUD operations using ORM framework (Hibernate)**	1

6.	**Week-6:ServletProgramming** Servlet Execution in tomcat. A Servlet program to print hello world. A Servlet program to display request details. A Servlet program to handle user form.	1
7.	**Week-7: Servlet Programming** Servlet program to create a cookie. A Servlet program to display cookie A Servlet program to do session tracking. Create a Basic Filter and Authenticate filter using Servlet Filters	2
8.	**Week-8:JSPProgramming** JSP program to display helloworld. JSP program to demonstrate arithmetic operations. JSP program to demonstrate jsp:forward action tag. JSP program to request implicit object, Perform CRUD operations using JSP	1
9..	**Week-9:JSTL Programs** Write a JSTL program to demonstrate core tags Write a JSTL program to find the given Number is Even OR Odd using CORE Tags in JSTL. Write a JSTL program to demonstrate IF Statement using Core Tags. Write a JSTL program to demonstrate Nested IF Statement Using Core Tags. Write a JSTL program to demonstrate XML tags to parse an XML document. Write aJSTLprogram to demonstrate XML tags to read an XML document	1
10.	**Week-10:** Introduction to MVC – Explore all the components with examples Introduction to Spring Framework – with examples	2
11.	**Week-11:** Implement MVC using Spring Framework.	2
12.	**Week-12:** Setting Up a Spring Boot and J2EE Hybrid Application	4

Change of Syllabus: 40%-Change in R23 Regulation compared to R20 Regulation

Name of the Unit	**Changes**
UNIT-3	JSP action elements, MVC Architecture with JSP, Servlets, and Spring MVC, CRUD operations with JSP. Introduction to Spring Boot and J2EE Integration.
UNIT-4	No change
UNIT-5	JSP action elements, MVC Architecture with JSP, Servlets, and Spring MVC, CRUD operations with JSP. Introduction to Spring Boot and J2EE Integration
Name Of The Experiment	**Changes**
Week-3	**JDBC Programs using Statement and Prepared Statement** 1) A program to test the connection with the database. 2) A program to create a table 3) A program to insert record in a table

	4) A program to fetch records from a table 5) A program to update record in a table 6) A program to delete record from a table
Week-4	**JDBC Programs using Callable Statement** 1) Create a store procedure which will insert one record in to employee table. 2) Create a store procedure which will retrieve salary for given employeeid. 3)Write a java application which will Call the above procedure and display appropriate information on screen.
Week5	Perform CRUD operations using ORM framework.
Week7	Create a Basic Filter and Authenticate filter using Servlet Filters
Week9	Perform CRUD operations using JSP

1.1.3 of NAAC

Name of the Course	**Course Code**	**Year of Introduction**	**Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development**	**Mapping with Employability/ Skill development / Entrepreneurship**
Advanced Java Programming Lab	IT327	R19	Request Processing in Web Application, Web Application Structure, Web Containers and Web Architecture Models, Client Server programming, Creating simple JDBC Application, Session Tracking: using Cookies, HTTP Session, JSP architecture, Tag Extension APL,Spring MVC,SpringBoot	Skill Development

INTERNET OF THINGS LAB

COURSE CODE: 23IT4221
L T P S
0 0 3 0

CREDITS 1.5
Sessional Marks: 50
End Exam: 3 Hours
End Exam Marks: 50

Prerequisite(s): Network Standards, Network Protocols, Network Technologies.

Course Objectives:

- Able to understand the application areas of IOT
- To acquire the knowledge and realize the revolution of Internet and Sensor Networks
- To acquire knowledge in designing, creating and deploying the real-time applications of IOT.

Course Outcomes:

After completion of this course, the students will be able to:

CO-1: Describe the embedded systems with Arduino environment and Raspberry Pi platforms.

CO-2: Analyze the designing, building and testing microcontroller based embedded Network Systems in modeling real world.

CO-3: Identify the appropriate cloud architectures and applications to deploy models in different domains.

CO-4: Create, Design and deploy IOT devices using Arduino and Raspberry Pi applications

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1	3	2	1	1	3				2	2		2	2	1
CO-2	3	2	1	1	3				2	2		2	2	1
CO-3	3	2	1	1	3				2	2		2	2	1
CO-4	3	2	1	1	3				2	2		2	2	1

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Course Outline

- **Introduction to Internet of things and Embedded Systems:** Defining IOT, Technologies and trends with IOT, Impact of IOT in society, components of an embedded system, core hardware components most commonly used in IoT devices, Interaction between software and hardware in an IoT device, use of networking and basic networking hardware, Structure of the Internet and meaning of a Network Protocol.
- **Arduino and RaspberryPi Programming:** Arduino Platform and C Programming: Composition of Arduino development board, read board schematics, Arduino IDE, what "shields" are and how they are used, Raspberry Pi platform and python programming: Composition of Raspberry pi development board, GPIO pins, Raspberry softwires.

- **Communication Protocols:** Protocol Standardization for IoT, M2M and WSN Protocols, SCADA and RFID Protocols, Unified Data Standards, Protocols – IEEE 802.15.4, BAC Net Protocol, Modbus, Zigbee Architecture.
- **IoT Physical Devices and Endpoints:** Basic building blocks of an IoT device, Exemplary device: Raspberry Pi, Raspberry Pi interfaces, Programming Arduino with sensor interfaces.
- **Interfacing with Arduino and Raspberry PI:** Sensors. Resistive Sensors, Actuators, Analog Actuators, Pulse Width Modulation,Arduino Libraries, I2C Communication, I2C Transactions, Arduino Shields, Ethernet Shield, Ethernet Library, Ethernet Client, Ethernet Server, Ethernet Shield Demo, WiFi Shield, Camera Module, pi camera Library, Capturing Images, PWM on RPI, Servo Control, Servo motor

References:

1. *The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World*.
2. A. Bahga and V. Madisetti, *Internet of Things – A Hands-on Approach*. Hyderabad, India: Universities Press, 2014. ISBN: 0-0996025510, 13: 978-0996025515.
3. H. Zhou, *The Internet of Things in the Cloud: A Middleware Perspective*. Boca Raton, FL, USA: CRC Press, 2012. ISBN: 978-1439892992.
4. D. Uckelmann, M. Harrison, and F. Michahelles, *Architecting the Internet of Things*. Berlin, Germany: Springer, 2011. ISBN: 978-3-642-19156-5.

List of Experiments

S. No	Experiments	CO
1	Introduction to IOT • Understanding IoT fundamentals • IOT Architecture and protocols • Various Platforms for IoT • Real time Examples of IoT • Overview of IoT components and IoT Communication Technologies • Challenges in IOT	1
2	Arduino Simulation Environment and Raspberry pi Environment • Arduino and Raspberry Pi Architecture • Setup the IDE, Writing Arduino Software, installing Raspberry OS • Arduino Libraries, Raspberry commands • Basics of Embedded C programming for Arduino and Raspberry pi • Interfacing LED, push button and buzzer with Arduino and Raspberry • Interfacing Arduino with LCD	1
3	Sensor & Actuators with Arduino & Raspberry Pi • Overview of Sensors working • Analog and Digital Sensors • Interfacing of Temperature, Humidity, Motion, Light and Gas Sensor with Arduino and Raspberry pi • Interfacing of Actuators with Arduino and Raspberry pi	2

	• Interfacing of Relay Switch and Servo Motor with Arduino and Raspberry pi	
4	Basic Networking with ESP8266 WiFi module • Basics of Wireless Networking • Introduction to ESP8266 Wi-Fi Module • Various Wi-Fi library • Web server- introduction, installation, configuration • Posting sensor(s) data to web server	2
5	IoT Protocols • M2M vs. IOT • Communication Protocols	3
6	Cloud Platforms for IOT • Virtualization concepts and Cloud Architecture • Cloud computing, benefits • Cloud services -- SaaS, PaaS, IaaS • Cloud providers & offerings • Study of IOT Cloud platforms • Thing Speak API and MQTT • Interfacing ESP8266 with Web services	3
7	Project	4

Change of Syllabus: Credits reduced to 1.5

1.1.3 of NAAC

Name of the Course	Course Code	Year of Introduction	Activities/Content with a direct bearing on Employability/ Entrepreneurship/ Skill development	Mapping with Employability/ Skill development / Entrepreneurship
Internet of Things Lab	23IT4221	2021	Technologies and trends with IOT, compositions of Arduino development boards, Shields, M2M and WSN Protocols, Raspberry Pi interfaces.	Employability

HIGH LEVEL REASONING, ADVANCED GRAMMAR AND INTERVIEW SKILLS

COURSE CODE: 23CR9104 | CREDITS 1
L T P S | Sessional Marks: 50
0 0 2 0 | End Exam: 3 Hours
End Exam Marks: 50

Prerequisites: Knowledge of LSRW Skills, Logical Reasoning Basics

Course Outcomes:

After completion of this course student will be able to:

CO-1: Apply advanced grammatical structures such as question tags, parallelism, and sign-post words to enhance clarity and coherence in written and spoken communication.
CO-2: Demonstrate proficiency in essay writing by structuring various types of essays effectively, expressing opinions logically, and engaging in formal discussions
CO-3: Utilize advanced grammar concepts like modifiers, subjunctive mood, redundancy, determiners, and word order for precision in writing and professional correspondence.
CO-4: Use logical thinking and analytical abilities to solve reasoning questions from syllogisms and cubes and dice and learning the skills to code and decode numbers into letters in crypt arithmetic and finding the solution
CO-5: Understand and solve puzzle related questions from specific and other competitive tests and able to solve the questions from critical reasoning and decision making

Mapping of Course Outcomes with POs and PSOs

COs/POs - PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2
CO-1									2	2		2		
CO-2									2	2		2		
CO-3									2	2		2		
CO-4	2													
CO-5	2													

Correlation levels 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

SECTION-A: HIGH LEVEL REASONING

UNIT - I:

Logical Deduction, syllogisms using Venn Diagrams

UNIT - II:

Cubes and cuboids- colouring – painting, seating arrangements – group reasoning

UNIT - III:

Crypt arithmetic, Data sufficiency

UNIT - IV:

Critical Reasoning – statements - conclusions, Critical Reasoning – statements - Assumptions, Critical Reasoning – statements - Implicit, Critical Reasoning – Decision Making, Critical Reasoning – statements, puzzle solving

UNIT - V:

Campus Readiness- company specific mock tests

SECTION-B: ADVANCED GRAMMAR & INTERVIEW SKILLS (AGIS)

UNIT - I:

Advanced Grammar – I: Question tags; Parallelism in structure; Sign-post words Self-introductions using professional formats

UNIT - II:

Essay-writing: getting used the various types of essays and learning how to write an effective essay

Expressing Opinions (along with asking, agreeing and disagreeing)

UNIT - III:

Advanced Grammar – II: Modifiers; Subjunctive mood; Redundancy; Determiners; Word-order

Resume-preparation (making it seem appropriate for the allied job-role)

UNIT - IV:

Sequencing; of sentences (para-jumbled); Eliminate the unrelated word from a group

Interview Preparation (practice on probable questions) along with the necessary etiquette

PySpark
(Skilled Course – MOOCS)

COURSE CODE: 23IT9304 | CREDITS 1
L T P S | Sessional Marks: 100
0 0 0 2 | End Exam: 3 Hours
End Exam Marks: -

Prerequisites: Python Programming, Java

Course Objectives:

This course demonstrates the usage of Apache Spark as a framework for building Machine Learning Models over standalone and distributed systems. In this course, the participant will learn how to load, transform the data present at varied locations like Hive Tables, Local server, HDFS flat files with different structures, and build machine learning models such as Linear Regression, Decision Tree, Random Forest, Gradient Boosting Machines, Logistic Regression, KMeans Clustering in Spark through Python interface. Large scale statistical model for solving issues like predicting heart attack, can be built using distributed frameworks like Apache Spark.

Course Outcomes:

CO-1: Understand how to perform ETL and analysis on data present in a distributed and standalone environment, through PySpark.
CO-2: Perform descriptive analysis on data using PySpark.
CO-3: Cleanse data and perform various ETL operations using PySpark.
CO-4: Perform predictive analytics with the help of machine learning algorithms, on a distributed system using PySpark

Topics

- Introduction to Spark
- Data used in the course
- Installation
- Spark Execution
- Basics Operations in RDD
- Handling Structured Data using RDD
- Basic Descriptive Statistics on RDD
- Basic Operations on Data Frame
- Data Frame Metadata
- Basic Descriptive Statistics on Data Frame
- Handling Semi-Structured Data
- Temp Tables
- Hive Integration
- User-Defined Function
- Optimizing Iterative Operations
- Spark Streaming
- Machine Learning using MLlib

MOOCs Platform

https://infyspringboard.onwingspan.com/web/en/app/toc/lex_3509975869549336000_shared/overview