



Shri Vile Parle Kelavani Mandal's

DWARKADAS J.SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Shri Vile Parle Kelavani Mandal's

Dwarkadas J. Sanghvi College of Engineering

(Autonomous College Affiliated to the University of Mumbai)

Scheme and detailed syllabus

Third Year B.Tech

in

Computer Science and Engineering (IoT and Cyber Security with Blockchain Technology) (Semester V)

Prepared by:- Board of Studies in Computer Science & Engineering

(IoT and Cyber Security with Blockchain Technology)

With effect from the Academic Year: 2026-2027



Scheme for Third Year Undergraduate Program in Computer Science and Engineering (IoT and Cyber Security with Blockchain Technology) : SEM V (Autonomous) Academic Year : 2024-25

Sr. no.	Course Code	Course	Teaching Scheme			Semester End Examination (SEE) - A						Continuous Assessment (CA) - B						Aggregate (A+B)	Credits Earned	
			Theory (Hrs)	Practical (Hrs)	Tutorial (Hrs)	Credits	Duration (Hrs)	Theory	Oral	Pract	Oral & Pract	SEE Total (A)	Term Test 1 (TT1)	Term Test 2 (TT2)	Term Test 3 (TT3)	Term Test Total (TT1 + TT2 + TT3)	Term Work			CA Total (B)
1	DS23BPC301	Advanced Data Structures and Algorithms	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
	DS23BPC301L	Advanced Data Structures and Algorithms Laboratory	-	2	-	1	2	-	-	-	25	25	-	-	-	-	25	25	50	1
2	DS23BPC302	Applied Cryptography	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
	DS23BPC302L	Applied Cryptography Laboratory	-	2	-	1	2	-	-	-	25	25	-	-	-	-	25	25	50	1
3	DS23BPC303	Introduction to Blockchain Technology	2	-	-	2	2	60	-	-	-	60	15	15	10	40	-	40	100	2
	DS23BPC303L	Introduction to Blockchain Technology Laboratory	-	2	-	1	2	-	25	-	-	25	-	-	-	-	25	25	50	1
4	DS23BPC304L	Web Application Development Laboratory	-	2	-	1	2	-	-	-	25	25	-	-	-	-	25	25	50	1
	DS23BPE301	Embedded System and IoT	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
5	DS23BPE301L	Embedded System and IoT Laboratory	-	2	-	1	2	-	25	-	-	25	-	-	-	-	25	25	50	1
	DS23BPE302	Distributed Computing	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
	DS23BPE302L	Distributed Computing Laboratory	-	2	-	1	2	-	25	-	-	25	-	-	-	-	25	25	50	1
	DS23BPE303	Computer Vision	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
	DS23BPE303L	Computer Vision Laboratory	-	2	-	1	2	-	25	-	-	25	-	-	-	-	25	25	50	1
	DS23BPE304	Cognitive Computing	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
6	DS23BPE304L	Cognitive Computing Laboratory	-	2	-	1	2	-	25	-	-	25	-	-	-	-	25	25	50	1
	DS23BMD301L	Artificial Intelligence	3	-	-	3	2	60	-	-	-	60	15	15	10	40	-	40	100	3
7	DS23BMD302L	Artificial Intelligence Laboratory	-	2	-	1	2	-	-	-	25	25	-	-	-	-	25	25	50	1
	DS23XHS131T	Environmental Studies	-	-	1	1	-	-	-	-	-	-	-	-	-	-	25	25	25	1
8	DS23XSC301P	Innovative Product Development III	-	2	-	1	2	-	-	-	25	25	-	-	-	-	25	25	50	1
	DS23BSC301	Artificial Intelligence for Cybersecurity	-	-	2	1	-	-	-	-	-	-	-	-	-	-	40	40	40	1
		Total	14	14	3	23	24	300	50	0	125	475	75	75	50	200	240	440	915	23

Prepared by

Checked by

Head of the Department

Vice Principal

Principal

(a)- Elective course

**Semester End Examination (SEE) – A**

Course	Assessment Tools	Marks	Time (hrs.)
Theory / * Computer based	Written paper based on the entire syllabus.	60	2
	* Computer based assessment in the college premises.		
Oral	Questions based on the entire syllabus.	25	As applicable
Practical	Performance of the practical assigned during the examination and the output / results obtained.	25	2
Oral & Practical	Project based courses - Performance of the practical assigned during the examination and the output / results obtained. Based on the practical performed during the examination and on the entire syllabus.	As per the scheme	2

Continuous Assessment (B):

Course	Assessment Tools	Marks	Time (mins)
Theory	a. Term Test 1 (based on 40 % syllabus)	15	45
	b. Term Test 2 (on next 40 % syllabus)	15	45
	c. Assignment / course project / group discussion / presentation / quiz/ any other.	10	--
	Total marks (a + b + c)	40	--
Audit course	Performance in the assignments / quiz / power point presentation / poster presentation / group project / any other tool.	--	As applicable
Laboratory	Performance in the laboratory and documentation.	25	
Tutorial	Performance in each tutorial & / assignment.	25	
Laboratory & Tutorial	Performance in the laboratory and tutorial.	50	



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)								T.Y.B.Tech		Semester: V	
Course: Advanced Data Structures and Algorithms								Course Code: DJS23BPC301			
Course: Advanced Data Structures and Algorithms Laboratory								Course Code: DJS23BPC301L			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	100
				60			15	15	10	40	
				Laboratory Examination			Term work				50
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Assignment		Total Term work	
				25	--	--	15	10	25		

Prerequisite:

1. Data Structures.
2. Design and Analysis of Algorithms.

Course Objectives

1. To provide conceptual and practical knowledge of Advanced Data Structures and Algorithms.
2. To cultivate algorithmic thinking and problem-solving skills through practice and exposure to diverse problem domains.
3. To develop strategies for breaking down complex problems into manageable sub-problems and applying suitable algorithms and data structures.
4. To develop skills to analyze problem complexity and choose appropriate data structures and algorithms for efficient solutions.

Course Outcomes

Students will be able to:

1. Implement advanced string data structures for efficient data processing.
2. Apply balanced search tree structures to optimize searching, indexing, and dynamic data management problems.



3. Design and implement heap-based data structures, disjoint set operations, and priority queue applications for efficient computation.
4. Evaluate hashing techniques and probabilistic data structures for secure, scalable, and high-performance data storage and retrieval.
5. Apply spatial data structures and graph-based algorithms to solve multidimensional searching, network flow, and matching problems.
6. Analyze and implement advanced graph algorithms for flow networks and maximum bipartite matching in real-world applications.

Detailed Syllabus:		
Unit	Description	Duration
	Data Structures for Strings	
1	Tries and Compressed Tries, Suffix Tree and Suffix Array, Bitwise Tries, Ternary Search Tree, Rope Data Structure, Introduction to Persistent Data Structures and their applications	06
	Balanced Search Trees	
2	Height Balanced and Weight Balanced Trees. Red-Black Tree, Splay Tree, AVL Tree, B Tree, B+ Tree, Skip List, Randomized BST, Tango Tree with applications, Industry Case Studies of Storage Engines.	10
	Heap and Operations	
3	Heap ordered Tree, Leftist Heap, Skew Heap, Tournament Tree. Binomial Heap vs Binary Heap, Pairing Heap, Double Ended Heap, Multidimensional Heaps. Treap with application, Disjoint Set Union, Top K elements, Two heaps-max heap and Min Hip.	06
	Hash Tables	
4	Universal Families of Hash Functions, Perfect Hash Functions, Cuckoo Hashing, Probabilistic Data Structures: Bloom filters, Count-Min Sketch, HyperLog. Locality Sensitive Hashing.	06
	Spatial Data Structures	
5	Introduction to Spatial Data and Multidimensional Searching, Interval Tree and its applications, Segment Tree for range query processing, Range Tree for multidimensional searching, Priority Search Tree and priority-based searching, KD Tree for partitioning multidimensional space, Quad Tree for two-dimensional spatial indexing, Octree for three-dimensional space representation, R Tree for spatial database indexing and nearest neighbor search.	08
	Graph Based Algorithms	
6	Flow Network Introduction: Residual Network, Augmenting Path, Ford-Fulkerson Method, Edmonds-Karp Method, Push-Relable Algorithm, Bipartite Matching: Maximum Bipartite Matching (using BFS/DFS approach)	06
Total		42



Instructions for Performing Advanced Data Structure Experiments on Competitive Programming Platforms

Objective: To implement, test, and analyze advanced data structures and algorithms using online coding platforms such as LeetCode, HackerRank, CodeChef, Codeforces, GeeksforGeeks Practice, or any equivalent competitive programming platform.

General Guidelines

1. Students should register and log in to any recognized competitive programming platform using their institutional or registered email ID.
2. The selected problems should be related to the topics covered in the course, including Tries, Suffix Structures, Balanced Search Trees, Heaps, Hashing Techniques, Spatial Data Structures, Disjoint Sets, and Graph Algorithms.
3. Students may use C, C++, Java, or Python for implementation, as permitted by the chosen platform.
4. Before implementation, students should study and document:
 - The problem statement
 - Underlying data structure or algorithm
 - Design approach
 - Expected time and space complexity
5. Students should implement and execute the solution using multiple test cases, including sample inputs and edge cases, to verify correctness and efficiency.
6. For application-oriented topics such as B-Trees, B+ Trees, Bloom Filters, Merkle Trees, KD Trees, and Graph Algorithms, students should analyze and document their practical relevance in databases, indexing systems, search engines, blockchain systems, recommendation systems, AI applications, or large-scale software systems.
7. For experiments involving database indexing, vector search, spatial indexing, or advanced tree structures, students should compare the performance of different approaches wherever applicable and summarize their findings.
8. Students are encouraged to attempt higher-difficulty problems and explore real-world applications of advanced data structures beyond the prescribed experiments.

Submission Instructions

For each experiment, submit:

- Problem title and platform name
- Data structure or algorithm used
- Time and space complexity
- Screenshots of successful submission and execution results
- Observations on performance and optimization



All source code, execution screenshots, complexity analysis, observations, and experiment write-ups must be maintained and submitted through Microsoft Teams for continuous assessment and evaluation.

List of Laboratory Experiments:	
Sr. No.	Suggested Experiments
1	Experiment on Amortized Analysis: Implement and compare Aggregate, Accounting, and Potential methods.
2	Implementation of Rope Data Structure. Perform String insertion, deletion, and concatenation. Compare with traditional string manipulation method
3	Implement persistent BST or persistent segment tree. Demonstrate versioning and rollback operations.
4	Experiment on Tries Data Structures: Implement Tries, Compressed Tries, and Bitwise Tries for string operations.
5	Implement nearest-neighbor search using KD Tree, Compare exact search and approximate nearest-neighbor search. Demonstrate applications in recommendation systems or semantic search.
6	Experiment on Advanced String Structures: Construct and analyze Suffix Tree, Suffix Array, and Ternary Search Tree.
7	Case Study on Database Storage Engines. Study indexing mechanisms used in modern database systems. Analyze performance implications of B-Tree/B+ Tree structures
8	Experiment on Balanced Search Trees: Implement AVL Tree and Red-Black Tree with insertion and deletion operations.
9	Implement Database Indexing using B-Tree and B+ Tree . Compare query execution time with and without indexes. Analyze insertion, deletion, and search performance.
10	Experiment on Heap Structures: Implement Binary Heap, Leftist Heap, and Skew Heap with operations.
11	Experiment on Advanced Heap Variants: Compare Binomial Heap, Pairing Heap, Double Ended Heap, and Tournament Tree.
12	Experiment on Hashing Techniques: Implement Cuckoo Hashing, Perfect Hashing, and Bloom Filters.
13	Experiment on Graph Algorithms: Implement Ford-Fulkerson, Edmonds-Karp, Push-Relabel, and Bipartite Matching algorithms.
14	Experiment on Spatial data structure.
15	Development of New Algorithm by Students Based on Any One Topic of Above-Mentioned Syllabus

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Minimum 10 experiments from the above suggested list or any other tutorial based on syllabus may be included, which would help the learner to understand topic/concept.



Books Recommended:

Text books:

1. Peter Brass, Advanced Data Structures, 1st Edition, Cambridge University Press, 2008.
2. Suman Saha, Shailendra Shukla, Advanced Data Structures Theory and Applications, 1st Edition, CRC Press and Taylor & Francis, 2020.
3. Harsh Bhasin, Algorithms Design and Analysis, Oxford, 1st Edition, 2015.

Reference Books:

1. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran Fundamentals of Computer Algorithms, 2nd Edition, Universities Press, 2008.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Introduction to Algorithms, 3rd Edition, The MIT Press, 2009.

Web resources:

1. <https://www.geeksforgeeks.org/dsa/dsa-tutorial-learn-data-structures-and-algorithms/>
2. <https://www.postgresql.org/docs/current/indexes-types.html#INDEXES-TYPES-BTREE>
3. [CS5800 Advanced Data Structures and Algorithms - YouTube](#)

Online Courses: NPTEL

1. Algorithmic Graph Theory and Data Structures By Prof. Sourav Mukhopadhyay | IIT Kharagpur https://onlinecourses.nptel.ac.in/noc26_cs08
2. Data Structures And Algorithms By Naveen Garg, IIT Delhi nptel.ac.in/courses/106102064

Evaluation Scheme:

Semester End Examination (A):

Theory:

1. Question paper based on the entire syllabus total comprising of 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

Oral examination will be based on the entire syllabus including the practical performed during laboratory sessions.



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Continuous Assessment (B):

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 can be taken as Assignment / course project / group discussion / presentation / quiz/ any other for 10 marks.

Laboratory: (Term work)

1. Term Work shall consist of at least 10 practical's based on the above list.
2. The distribution of marks for term work shall be as follows:
 - i. Laboratory work (Performance of Experiments, Write-up): 15 Marks
 - ii. Assignment: (HackerRank Quiz, Descriptive/Analytical Questions): 10 Marks.

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work and upon fulfilling minimum passing criteria in the term work.

Prepared by

Checked by

Head of the Department

Vice-Principal

Principal



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)								T.Y.B.Tech		Semester : V	
Course : Applied Cryptography								Course Code: DJS23BPC302			
Course: Applied Cryptography Laboratory								Course Code: DJS23BPC302L			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	100
				60			15	15	10	40	
				Laboratory Examination			Term work				
3	2	—	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project presentation Assignment		50	
				—	--	25	15	10	25		

Pre-requisite:

1. Mathematics – I
2. Mathematics – II
3. Fundamentals of Sensor and Secured Technologies
4. Computer Networks

Course Objectives: The objective of the course is

1. To introduce classical encryption methods, along with key concepts in modular arithmetic and number theory.
2. To gain a foundational understanding of cryptography and its core principles.
3. To explore the functioning and applications of various cryptographic algorithms
4. To gain skills in using cryptographic tools for secure communication.

Course Outcomes: On completion of this course, learner will be able to

1. Apply the concepts of number theory to solve cryptographic problems
2. Apply classical encryption and decryption techniques
3. Implement symmetric-key encryption algorithms, modes of operation, and key distribution mechanisms.
4. Apply public-key cryptographic algorithms for secure communication and authentication.
5. Demonstrate cryptographic hash functions in various security context.
6. Implement digital signature algorithms and understand digital certificates within a PKI infrastructure

**Detailed Syllabus**

Unit	Description	Duration
1	Number Theory for Security and Cryptography: OSI Security Architecture, Modular Arithmetic, Euclidean Algorithm, Prime Numbers, Relatively Prime Numbers, Primitive Roots, Fermat's Little Theorem, Euler Totient Function, Extended Euclidean Algorithm, Chinese Remainder Theorem	07
2	Fundamentals of Cryptography: Introduction, plain text and cipher text, Classical Encryption techniques, Symmetric cipher model, mono-alphabetic and polyalphabetic substitution techniques: Vigenere Cipher, Playfair Cipher, Hill Cipher, Affine Cipher, Transposition Techniques: keyed and keyless transposition ciphers.	07
3	Symetric-Key Encryption: Block Ciphers, Stream Ciphers, Modes of Operation, Feistel Ciphers, Data Encryption Standard (DES), Cracking DES, Triple DES, Advanced Encryption Standard (AES), Modern Block Cipher, RC5, Cryptanalysis, Weak Keys. Symetric-Key Distribution: KDC, Needham-Schroeder Protocol, Diffie-Hellman Key Exchange Algorithm	10
4	Public-Key Cryptography: Public-Key Cryptography, Knapsack Cryptosystem, RSA Cryptosystem, Attack on RSA, ElGamal cryptosystem, Security of ElGamal, Elliptic Curve Cryptography [ECC]	06
5	Cryptographic Hash Functions: Cryptographic Hash Functions – MD5, attack on MD5, SHA-1, SHA-3, SHA-256, SHA-512 MAC, HMAC, CMAC	06
6	Digital Signature Schemes and Digital Certificates: Digital Signature – Process, Services, Attacks on Digital Signature, Digital Signature Schemes – RSA, El Gamal, Digital certificate, Chain of certificate, PKI, Quantum Cryptography: Definition, Basic Principles of Quantum Mechanics Relevant to Cryptography, Applications and Challenges	06
Total		42

List of Laboratory Experiments:

Sr. No.	Suggested Experiments
1	Implement the Caesar Cipher Tech using socket programming



2	Implement Columnar Transposition Technique using socket programming.
3	Implement Vigenère Cipher Technique using socket programming.
4	Implement Affine Cipher techniques.
5	Implement own encryption technique which consist of novel substitution and Transposition approach.
6	Implement Playfair Cipher Technique.
7	Implement Hill Cipher Technique
8	Implementation of Simplified DES Encryption and decryption
9	Implement simplified AES DES Encryption and decryption.
10	Implement DES key generation Techniques.
11	Implement AES Key Generation Techniques.
12	Implementation and analysis of RSA crypto system
13	Implement Knapsack Cryptosystem.
14	Implementation of Diffie-Hellman Key exchange algorithm
15	Implementation of Message digest using MD5/SHA-1
16	Implementation of Digital Signatures in Cryptography

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Books Recommended:

Text Books:

1. William Stallings, Cryptography and Network Security, Principles and Practice, 8th Edition, Pearson Education, June 2023.
2. Behrouz A. Ferouzan, —Cryptography & Network Security, Tata McGraw Hill 3rd Edition 2015
3. Bernard L. Menezes — Cryptography, Network Security, and Cyber Laws, Cengage Learning 2018
4. Network Security Bible, Eric Cole, Second Edition, Wiley. 2015

Reference Books:

1. Applied Cryptography, Protocols Algorithms and Source Code in C, Bruce Schneier, Wiley 2015
2. Cryptography and Network Security, Atul Kahate, Tata Mc Graw Hill, Fourth Edition 2019



Web Resources:

1. Data Encryption standard: <https://www.geeksforgeeks.org/data-encryption-standard-deset-1>
2. Advance Encryption standard:
https://www.tutorialspoint.com/cryptography/advanced_encryption_standard.htm
3. Digital Signature: <http://www.javatpoint.com/java-digital-signature>
4. Challenge Response Protocols:
<https://www.tutorialspoint.com/challengeresponseauthentication-mechanism-cram>.

Online Courses: NPTEL / Swayam

1. Foundations of Cryptography, Prof. Ashish Choudhury, IIIT Bangalore
<https://nptel.ac.in/courses/106106221>.
2. Cryptography I Dan Boneh, Sanfort
<https://www.coursera.org/learn/crypto>.
3. Introduction to Applied Cryptography Specialization Sang-Yoon Chang, University of Colorado
<https://www.coursera.org/specializations/introduction-applied-cryptography>

Evaluation Scheme:

Continuous Assessment (A)

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 can be conducted as Assignment / course project / group discussion / presentation / quiz / any other for 10 marks.

Laboratory: (Term work)

Term work shall consist of minimum 8 experiments and a Mini project.

The distribution of marks for term work shall be as follows:

- i. Laboratory work (Performance of Experiments): 15 Marks
- ii. Mini Project: 10 marks

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work.

Semester End Examination (B):

Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

Oral examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.

Prepared by

Checked by

Head of the Department

Vice Principal

Principal



Program: B.Tech in Computer Science and Engineering (IoT and Cybersecurity with Block chain Technology)							T.Y.B.Tech		Semester : V		
Course: Introduction to Blockchain Technology							Course Code: DJS23BPC303				
Course: Introduction to Blockchain Technology Laboratory							Course Code: DJS23BPC303L				
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	
				60			15	15	10	40	
				Laboratory Examination			Term Work				
2	2	--	3	Oral	Practical	Oral & Practical	Laborat ory Work	Tutorial / Mini project /presentation/ Journal/ Practical		Total Term work	
				25	—	—	15	10	25		
				100							
				50							

Prerequisite:

1. Fundamentals of Sensor and Secured Technologies
2. Computer Networks

Course Objectives: The objective of the course is

1. To understand emerging Blockchain Technology and its relevance with cryptography.
2. To demonstrate the use of cryptography required for Blockchain.
3. To understand smart contracts, wallets, and consensus protocols.
4. To design and develop Blockchain applications.

Course Outcomes: On completion of the course, learner will be able to:

1. Acquire basic knowledge of Blockchain technology to implement cryptographic primitives useful for Blockchain
2. Analyze the Bitcoin network and compare blockchain consensus mechanisms
3. Design and deploy decentralized applications on Ethereum using Solidity
4. Illustrate the architecture, transaction flow, and chaincode execution in Hyperledger Fabric.
5. Evaluate decentralized finance platforms and blockchain-based transaction mechanisms.

**Detailed Syllabus:**

Unit	Description	Duration
1	Basic Crypto primitives for Blockchain Technology Preview: Introduction to Blockchain Technology The Model of Decentralization, Block structure, Block header, Types of Blockchain: Public, Private and Hybrid Blockchain. Cryptographic Primitives, Cryptographic Hash, Hash Functions, Merkle Tree, Hash Chain, Construction of Chain of Blocks, Digital Signature	05
2	Bitcoin and Consensus Bitcoin P2P network, Transaction flooding in Bitcoin mining, double spending attack, Forks, The Monopoly Problem-51% attack Consensus: Consensus Approach, Consensus Algorithms: Proof-of-Stake (PoS), Proof-of-Work (PoW), Proof-of-Burn (PoB), Proof-of-Elapsed Time (PoET), Proof of History (PoH), State Machine Replication as a Consensus, Crash Fault Tolerance, PAXOS, Byzantine Fault Tolerant (BFT), BFT Consensus, Practical BFT	07
3	Ethereum Introduction to Ethereum and its Components, Ethereum Architecture and Ecosystem Ethereum Accounts (Externally Owned Accounts and Contract Accounts), Ethereum Virtual Machine (EVM), Ethereum Transactions and Gas Mechanism, Ether, Gas, and Gas Fees, Comparison between Bitcoin and Ethereum, Ethereum Test Networks and Development Environment Smart Contracts: Concepts, Features, and Applications Introduction to Solidity Programming Deploying and Testing Smart Contracts using Remix IDE	07
4	Hyperledger Introduction to Hyperledger Fabric, Key features of Hyperledger fabric, Hyperledger Fabric Architecture, Ethereum v/s Hyperledger framework, Fabric Transaction Flow, Hyperledger Tools and Libraries, Hyperledger Fabric Chaincode	05
5	Decentralized Finance & Smart Transactions Blockchain in DeFi: Case Study on any of the Blockchain Platforms, Smart property, Efficient micro-payments, Coupling transactions and payments, Escrow transactions	04
Total		28

List of Laboratory Experiments:

Sr. No.	Suggested Experiments
1	To create basic Blockchain with sample transactions and print it.
2	To implement Merkle root from the transactions and verify the validity of transactions using it.
3	To implement Proof of Work (PoW) concept in Bitcoin Mining and demonstrating it.
4	To analyse and implement Unspent Transaction Outputs (UTXOs) in Bitcoin and demonstrate the transactions using UTXOs.
5	To create and deploy Smart Contract using Solidity and Remix IDE.



6	To perform Embedding wallet and transaction using Solidity and MetaMask.
7	To implement blockchain using Geth (Go-Ethereum).
8	To implement local Blockchain using tools viz. Ganache and Hardhat.
9	To interacting with the Ethereum Blockchain Using Web3.js
10	Developing and Deploying a Permissioned Blockchain Network using Hyperledger Fabric
11	To install Hyperledger Fabric and demonstrate its usability.
12	To query and invoke transactions on Fabric Test Network.
13	To develop an Ethereum smart contract that locks funds from a buyer to design a secure escrow system.
14	Deploy an Ethereum smart contract representing ownership of an asset and Implement a function to transfer ownership securely using a blockchain transaction.
15	Mini Project (A group of 3 to 4 students is required to develop an application using blockchain technology and submit report).

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Books Recommended:**Text Books:**

1. Imran Bashir , Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more, 3rd Edition, Packt Publishing, 2020, ISBN: 9781839213199,
2. Kumar Saurabh , Ashutosh Saxena, Blockchain Technology: Concepts and Applications , 1st Edition, Wiley Publication, 2020, ISBN:978-81-265-5766-0
3. S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, -Blockchain Technology: Cryptocurrency and Applications, Oxford University Press, 2019
4. Cryptography and Network Security – Principles and Practice by William Stallings, Pearson 2017

Reference Books:

1. Antony Lewis, Basics of Bitcoins and Blockchain, Mango Publishing, 2021
2. Yathish R and Tejaswini N, Blockchain for Beginners, Shroff/X-Team, 2019
3. Daniel Drescher, Blockchain Basics, A non-Technical Introduction in 25 Steps, Apress, 2017.
4. Luc Desrosiers, Nitin Gaur, Salman A. Baset, Venkatraman Ramakrishna, Blockchain with Hyperledger Fabric, Packt Publishing, 2020
5. Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions, Apress, 2018



Web resources:

1. Hyperledger Tutorials - <https://www.hyperledger.org/use/tutorials>
2. Ethereum Development Resources - <https://ethereum.org/en/developers/>
3. Solidity Tutorial- <https://www.tutorialspoint.com/solidity/index.htm>
4. Metamask- <https://docs.metamask.io/guide/>

Online Courses: NPTEL / Swayam

1. Blockchain and its Applications, By Prof. Sandip Chakraborty, Prof. Shamik Sural IIT Kharagpur
https://onlinecourses.nptel.ac.in/noc23_cs47/preview
2. Blockchain Architecture Design and Use Cases, By Prof. Sandip Chakraborty & Dr. Praveen Jayachandran | IIT Kharagpur and IBM,
https://onlinecourses.nptel.ac.in/noc19_cs63/preview
3. Blockchain, By Dr. Mayank Aggarwal, Gurukul Kangri Vishwavidyalaya, Haridwar
https://onlinecourses.swayam2.ac.in/aic21_ge01/preview

Evaluation Scheme:

Semester End Examination (A):

Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

Oral examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.

Continuous Assessment (B)

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 can be taken as Assignment / course project / group discussion / presentation / quiz / any other for 10 marks.

Laboratory: (Term work)

Term work shall consist of minimum 8 experiments and a Mini project.

The distribution of marks for term work shall be as follows:

- i. Laboratory work (Performance of Experiments, Write-up): 10 Marks
- ii. Assignments (HackerRank Quiz, Descriptive/Analytical Questions): 5 Marks
- iii. Miniproject (Implementation, Report): 10 marks

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work.

Prepared by

Checked by

Head of the Department

Vice Principal

Principal



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)							T.Y.B.Tech		Semester : V		
Course : Web Application Development Laboratory							Course Code: DJS23BPC304L				
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	50
				--			--	--	--	--	
				Laboratory Examination			Term work				
--	2	--	1	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Assignment		Total Term work	50
				--	--	25	15	10		25	

Prerequisite:

1. Object Oriented Programming using Java
2. Python programming

Objectives: The objectives of the course are:

1. To orient students to Web Programming fundamental
2. To develop hands-on skills in building dynamic and interactive web applications using modern web development technologies.
3. To enhance problem-solving abilities and encourage creativity and innovation in designing and implementing web applications
4. To Work collaboratively on web development projects to enhance teamwork, communication, and project management skills

Outcomes: On completion of the course, learner will be able to:

1. Design and develop responsive and user-friendly web applications.
2. Build dynamic and interactive web applications.
3. Explore new web development technologies.
4. Work effectively as a member of a team to complete mini project.



Detailed Syllabus:		
Unit	Description	Durati on
	Web Programming Fundamentals :	
1	Introduction to Web Programming, Installation of IDE, Introduction to basic structure of a website, Title, Script, Link & meta tags. Understanding of headings, paragraphs. Image and Anchor tags, Understanding Lists & Tables, Forms and Input tags, PHP connection code to db, Local server setup and uses, Inline and block elements, Ids and Classes concept, Working of web browser, XML introduction, HTTP protocol, Json introduction	04
	Static web page design –HTML, CSS and CSS3 :	
2	HTML entities and semantic tags. HTML Media, Video, Audio, Plugins. HTML API's (Geolocation, Web Storage, SSE, etc) Concepts of CSS: Introduction to CSS, Inline, Internal and External CSS, Selectors, Developer tools in chrome, CSS Box model, margin, padding, fonts, colors, Borders, backgrounds, Float and clear, links, buttons Creating Navigation menu, display property, positions (absolute, relative, fixed & sticky), visibility, z-index, flexbox, web units, media queries, pseudo selectors, shadow properties, Introduction to animation and key frames, responsive properties, Introduction to bootstrap 4 & 5	05
	Client side scripting – JavaScript :	
3	Introduction to Javascript (Frontend & backend), writing in-browser javascript & developer console. Variables, Data types, Operators, String and String functions, scope, conditional statements, functions, loops, DOM library functions, Event Listeners, arrow functions.	04
	NodeJs :	
4	NodeJs introduction and installation, First app, Asynchronous programming, Callback concept, Event loops, REPL, Event emitter, Networking module, Buffers, Streams, File system, Web module. Serving HTML files using NodeJs, Node package manager.	04
	Introduction to Angular :	
5	Angular Development Environment, Basic Angular Component and Template, Data Binding and Event Handling, Fetching Data from APIs and Displaying using HTTP Client, Routing and Navigation, Forms, Form Validation, Authentication and Authorization, Testing Angular Components and Services, Implementing in Angular Applications .	05



6	Introduction to ReactJs and Advance React : Introduction and Installation, understanding JSX, Prop & Prop Types, Understanding State and Event Handling, TextUtils, Functional components- Refs, Use effects, Hooks, Flow architecture, Model-View Controller framework, Flux, Bundling the application.	04
Total		26

List of Laboratory Experiments:

Sr. No.	Suggested Experiments
1	HTML, HTML5: a. Create a Basic Web Page Using HTML and HTML5 tags . b. Incorporate Multimedia Elements with HTML5 (e.g., Audio, Video) . c. Build a Navigation Menu using HTML5 Semantic Elements .
2	XML: a. Create XML Documents and Validating XML Syntax . b. Create XML-based Web Services. c. Convert XML to JSON and vice versa .
3	CSS: a. Enhance User Interfaces with CSS Transitions and Animations . b. Create Responsive Layouts with CSS Grid.
4	CSS3: a. Design a responsive web page using media queries and CSS3 . b. Implement CSS3 Filters and Effects for Visual Enhancements .
5	Bootstrap: a. Build a Responsive Layout with Bootstrap Grid System . b. Style Buttons and Forms using Bootstrap Components . c. Implement Bootstrap Dropdowns and Accordions for Content Organization .
6	JavaScript: a. Create Interactive Web Elements with JavaScript Event Handling . b. Implement Form Validation using JavaScript . c. Build Dynamic Content with JavaScript DOM Manipulation .
7	Program to design a calculator using JavaScript.
8	NodeJs: a. Set up a Node.js Development Environment . b. Create and Running a Simple Node.js Server . c. Build a RESTful API with Node.js and Express .
9	Work with Databases in Node.js (e.g., MongoDB, MySQL) .



10	Angular: a. Create Dynamic and Responsive User Interfaces with Angular Directives. b. Build Forms and Performing Form Validation in Angular . c. Deploy an Angular Application to a Web Server .
11	ReactJs: a. Set up a React Development Environment . b. Implement Component State and Handling User Interactions in React . c. Fetch Data from APIs and Displaying it in React . d. Test React Components and Hooks .
12	Advance React: a. Build React Components with Flux Data Flow . b. Implement Model-View-Controller in React with State Management Libraries . c. Implement Controllers for Handling User Interactions in React MVC.
13	Mini Project – (A group of 3 to 4 students is required to develop Complete website development using client and server side technologies.

Any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt

Books Recommended:

Text Books:

1. Haverbeke, M, Eloquent JavaScript: A Modern Introduction to Programming, 3rd Edition, No Starch Press, 2021.
2. Chinnathambi, C., Learning React: Modern Patterns for Developing React Apps, Packt Publishing, 2023.
3. DT Editorial Services, HTML5 Black Book, 2nd Edition, Dreamtech Press, 2016.
4. Ben Frain, Responsive Web Design with HTML5 and CSS3, 2nd Edition, Packt Publishing, 2015.
5. Casciaro, M., and Lavin, M., Node.js Design Patterns, 2nd Edition, Packt Publishing, 2022
6. Stoyan Stefanov, React Up Running Building Web Applications, 1st Edition, O'Reilly Media Inc., 2016.
7. David Sklar, Learning PHP 5, 1st Edition, O'Reilly Media Inc., 2004.

Reference Books:

1. Ho, A., and Prieto, A., Fullstack React: The Complete Guide to ReactJS and Friends, 2nd Edition, Fullstack.io, 2020
2. Benjamin LaGrone, HTML5 and CSS3 Responsive Web Design Cookbook, 1st Edition, Packt Publishing, 2013.
3. Christopher Schmitt, Kyle Simpson, HTML5 Cookbook, 1st Edition, O'Reilly Media Inc., 2011.
4. Uttam K. Roy, Web Technologies, 1st Edition, Oxford University Press, 2010.
5. Greg Sidelnikov, React. Js Book: Learning React JavaScript Library from Scratch, 1st Edition, Independently Published, 2017.



6. Luke Welling; Laura Thomson, PHP and MySQL Web Development, 5th Edition, Addison-Wesley Professional PTG, 2017.

Web resources:

1. Basics of HTML
<https://html-iitd.vlabs.ac.in/exp/introduction-to-html/references.html>
2. Master Django Web Development: Hands-On Projects
<https://html-iitd.vlabs.ac.in/exp/introduction-to-html/references.html>
3. Web Development courses for Students by IBM
<https://skillsbuild.org/students/course-catalog/web-development>
4. Namaste JavaScript Online Available:
<https://www.youtube.com/playlistlist=PLlasXeu85E9cQ32gLCvAvr9vNaUccPVNP>
5. Web Development Basics: Introduction to HTML5, CSS, and JavaScript
<https://www.coursera.org/learn/html-css-javascript-for-web-developers>
6. Tutorial Tic Tac toe <https://reactjs.org/tutorial/tutorial.html>
7. React Redux <https://react-redux.js.org/introduction/quick-start>
8. <https://developer.mozilla.org/en-US/>

Online Courses:

1. Introduction to Modern Application Development - Prof. Soumya Kanti Ghosh, IIT Madras
Course link: [Introduction to Modern Application Development - Course](#)
2. Modern Application Development Prof. Aamod Sane, Prof. Abhijat Vichare, Prof. Madhavan Mukund
Course link: [Modern Application Development - Course](#)

Evaluation Scheme:

Continuous Assessment (A):

Laboratory:

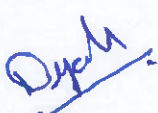
1. Term work shall consist of minimum 8 experiments.
2. The distribution of marks for term work shall be as follows:
 - i. Laboratory work (Performance of Experiments): 15 Marks
 - ii. Mini Project (Implementation and Report): 10 marks

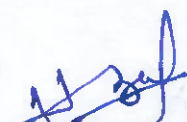
The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work

Semester End Examination (B):

Laboratory:

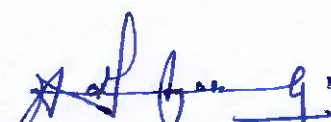
Oral and Practical examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.


Prepared by


Checked by


Head of the Department


Vice-Principal


Principal



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)								T.Y.B.Tech		Semester: V	
Course: Embedded System and IoT								Course Code: DJS23BPE301			
Course: Embedded System and IoT Laboratory								Course Code: DJS23BPE301L			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	100
				60			15	15	10	40	
				Laboratory Examination			Term work				
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project presentation Assignment		50	
				25	--	--	15	10	25		

Prerequisite:

1. Basics of Programming
2. Fundamentals of Sensor and Secured Technologies

Course Objectives: The objective of the course is

1. To emphasis on working of Microcontroller.
2. To Understand the definition and significance of the Internet of Things
3. To provide hands-on experience in designing IoT applications using various sensors, actuators, and cloud-based platforms.
4. To familiarize with IoT communication protocols, data processing techniques, edge analytics, and IoT security challenges.

Course Outcomes: On completion of the course, learner will be able to:

1. Describe the architecture of 8051 microcontrollers.
2. Implement interfacing with ARM, ESP32 and Raspberry Pi.
3. Analyse applications in areas of IoT using sensors and actuators.
4. Apply IoT communication models and protocols.
5. Utilize Edge analytics to perform data stream mining.
6. Apply various applications and security in IoT.



Detailed Syllabus:		
Unit	Description	Duration
1	The Microcontroller Architecture: Introduction to 8051 Microcontroller, Architecture, Pin configuration, Memory organization, Input /Output Ports, Counter and Timers, Serial communication, Interrupts.	05
2	Architecture: Basic ARM architecture, ARM organization core Data flow Model, ARM register organization, current program register organization. Introduction to ESP32, Architecture of ESP32, Introduction to Raspberry Pi, Understanding SoC architecture and SoCs used in Raspberry Pi, Pin Description of Raspberry Pi, On-board components of Rpi.	08
3	IoT with Sensors and actuators: IoT Growth- A Statistical View, Application area of IoT, Things in IoT, IoT stack, IoT challenges, Cyber physical systems versus IoT, Wireless sensor Network with IoT, Sensors for different IoT applications, working principle of Soil Moisture Sensor, Gas Sensor, Obstacle Sensor, Gyro Sensor, LDR Sensor, GPS Sensor Medical Sensor: Heartbeat & Pulse Sensor Actuators: Motors – Servo, DC, Stepper; Relay – SPDT, DPDT, Solenoid	09
4	IoT model and protocols IoT Reference Model , IoT Levels, Various Operating System TinyOS, Contiki OS, RTOS, Protocol Classification, MQTT, XMPP, DDS, AMQP, CoAP, REST, 6LoWPAN. IoT Routing Protocols, Data-centric and Flat-Architecture Protocols, Flooding, Gossiping, SPIN, SPIN PP, SPIN EC, SPIN BC, Hierarchical Protocols, LEACH, QoS-Based Protocols.	08
5	Edge Analytics: Near Real Time Sensor Stream Processing, Introduction, Streaming Data, Data stream management systems, Edge Analytics. Overview of Edge computing, Cloud computing, Fog computing.	06
6	Applications and Security: IoT applications: Smart Agriculture: Precision farming, soil monitoring, weather prediction, Smart Healthcare: Remote patient monitoring, wearable sensors, AI diagnostics, Smart Cities: Traffic management, waste management, smart lighting, Industrial IoT (IIoT): Predictive maintenance, automation, robotics, Smart Homes: Home automation, energy management, security systems. IoT Security: IoT Threats and Attack Vectors, Blockchain and IoT Security.	06
Total		42



List of Laboratory Experiments:	
Sr. No	Suggested Experiments
1	To study an overview of personal computer (PC) assembly, and its various components.
2	To study and explore the Tinkercad Simulator Case study based on current trends and advancement in IoT.
3	To implement Motion Detection and Alert System Using PIR Sensor in IoT
4	To implement interfacing of ultrasonic sensor with ESP32.
5	To monitor a weather using ESP32 and Think speak.
6	To implement interfacing of LM35 with Arduino Uno and LCD monitor.
7	To control home devices (LED) using self-hosted page on Amazon AWS.
8	To send email alert using ESP32 via SMTP server.
9	To monitor heart rate using Pulse sensor and Arduino Uno.
10	To implement program for ESP32 DHT11/DHT22 Temperature and Humidity Web Server with Arduino IDE.
11	To perform Soil Moisture Monitoring and Analysis Using IoT Sensors
12	To study and implement IoT Data processing using Pandas
13	Mini project (A group of 3 to 4 students is required to develop an application using IoT sensors and devices and submit report).

Any other experiment may be included, which would help the learner to understand the topic/concept.

Books Recommended:

Text books:

1. Shriram Vasudevan, Abhishek Nagarajan, RMD Sundaram, Internet of Things, Wiley Publication, Second Edition, 2020.
2. Surya Durbha, Jyoti Joglekar, Internet of Things, Oxford University Press, First Edition, 2021.
3. M. A. Mazidi, J. G. Mazidi and R. D. Mckinlay, The 8051 Microcontroller & Embedded systems, Pearson Publications, Second Edition 2006.
4. C. Kenneth J. Ayala and D. V. Gadre, The 8051 Microcontroller & Embedded system using assembly & 'C', Cengage Learning, Edition 2010.
5. Brojo Kishore Mishra and Amit Vishwasrao Salunkhe, Internet of Things: Technological Advances and New Applications, CRC 2024.



Reference Books:

1. RFID and the Internet of Things, by Herve Chabanne, Wiley publication, 2011 .
2. Embedded system design A Unified hardware/software Introduction, Frank Wahid, Tony Givargis, Wiley, 2006 .
3. Embedded System Architecture, Programming and Design, Raj Kamal, McGraw Hill, 2017.

Web resources:

1. Embedded Systems Academy- <https://www.embedded-sys.com/plus/>
2. Embedded Systems Basics by Tutorials point-
https://www.tutorialspoint.com/embedded_systems/index.htm
3. Embedded Systems Programming Course by Udemy-
<https://www.udemy.com/course/introduction-to-embedded-systems/>
4. Course on- Introduction to Embedded Systems Software and Development Environments-
<https://www.coursera.org/learn/introduction-embedded-systems>

Online Courses: NPTEL/SWAYAM

1. NPTEL: Prof. Prabhakar, IISc Bangalore, Design for Internet of Things,
https://onlinecourses.nptel.ac.in/noc21_ee85/preview
2. NPTEL: Sudip Misra, IIT Khargpur, Introduction to IoT: Part-1,
<https://nptel.ac.in/courses/106/105/106105166/>



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Evaluation Scheme:

Continuous Assessment (A)

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 can be conducted as Assignment / course project / group discussion / presentation / quiz/case study for 10 marks.

Laboratory: (Term work)

Term work shall consist of minimum 8 experiments and a Mini project.

The distribution of marks for term work shall be as follows:

- i. Laboratory work (Performance of Experiments, Write-up): 15 Marks
- ii. Mini project (Implementation, Report): 10 marks

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work.

Semester End Examination (B):

Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

Oral examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.

Prepared by

Checked by

Head of the Department

Vice Principal

Principal



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)							T.Y.B.Tech		Semester : V		
Course : Distributed Computing							Course Code: DJS23BPE302				
Course: Distributed Computing Laboratory							Course Code: DJS23BPE302L				
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)			Total marks (A+ B)	
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	100
				60			15	15	10	40	
				Laboratory Examination			Term work			Total Term work	
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project presentation Assignment		25	50
				25	--	--					

Prerequisite:

1. Operating systems.
2. Computer Network.

Course Objectives: The objective of the course is

1. To provide students with contemporary knowledge in distributed systems
2. To empower students with the skills required to analyze, design, and develop distributed applications.
3. To develop expertise in assessing the performance of distributed synchronization algorithms.

Course outcomes: On completion of this course, learner will be able to:

1. Demonstrate knowledge of the basic elements and concepts related to distributed system.
2. Illustrate the middleware technologies that support distributed applications such as RPC, RMI and Object based middleware.
3. Analyze the various techniques used for clock synchronization and mutual exclusion.
4. Showcase the concepts of Resource and Process management and synchronization algorithms.
5. Demonstrate the concepts of Consistency and Replication Management.
6. Apply knowledge of Distributed File Systems to analyze NFS, AFS, and build large-scale distributed applications.



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Detailed Syllabus:		
Unit	Description	Duration
1	Introduction to Distributed Systems Characterization of Distributed Systems: Issues, Goals, and Types of distributed systems, Distributed System Models, Hardware concepts, Software Concept. Middleware: Models of Middleware, Services offered by middleware, Client Server model.	06
2	Communication: Layered Protocols, Interprocess communication (IPC): MPI, Remote Procedure Call (RPC), Remote Object Invocation, Remote Method Invocation (RMI). Message Oriented Communication, Stream Oriented Communication, Group Communication.	07
3	Synchronization : Clock Synchronization, Logical Clocks, Election Algorithms, Mutual Exclusion, Distributed Mutual Exclusion-Classification of mutual Exclusion Algorithm, Requirements of Mutual Exclusion Algorithms, Performance measure. Non Token based Algorithms: Lamport Algorithm, Ricart-Agrawala's Algorithm, Maekawa's Algorithm. Token Based Algorithms: Suzuki-Kasami's Broadcast Algorithms, Singhal's Heuristic Algorithm, Raymond's Tree based Algorithm.	08
4	Resource and Process Management : Desirable Features of global Scheduling algorithm, Task assignment approach, Load balancing approach, load sharing approach Introduction to process management, process migration, Threads, Virtualization, Clients, Servers, Code Migration	07
5	Consistency, Replication and Fault Tolerance: Introduction to replication and consistency, Data-Centric and Client-Centric Consistency Models, Replica Management. Fault Tolerance: Introduction, Process resilience, Reliable client-server and group communication, Recovery	07
6	Distributed File Systems and Name Services: Introduction and features of DFS, File models, File Accessing models, File-Caching Schemes, File Replication, Case Study: Distributed File Systems (DSF), Network File System (NFS), Andrew File System (AFS) Introduction to Name services and Domain Name System, Directory Services, Case Study: The Global Name Service, The X.500 Directory Service	07
Total		42



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



List of Laboratory Experiments:	
Sr. No.	Suggested Experiments
1	Implementing Remote Procedure Calls (RPC) and Remote Method Invocation (RMI) in Client-Server Systems
2	Implementing IPC Techniques for Efficient Process Coordination
3	Implementing Group Communication for Collaborative Systems
4	Designing Efficient Load Balancing Strategies for Scalability
5	Implementing Name Resolution Protocols for Network Communication
6	Implementing and Analyzing Election Algorithms in Distributed Systems
7	Implementing and Analyzing Clock Synchronization Algorithms
8	Implementing Deadlock Detection and Recovery in Distributed Systems
9	Designing and Implementing a Distributed File System
10	Case Study on a system that provides a global naming infrastructure, allowing unique identification of resources across a distributed network.
11	Case Study on distributed file system designed to handle large-scale data storage and retrieval for social media platforms like Facebook, ensuring data availability, fault tolerance, and fast access.
12	Case study analyzing how Google designs and scales its distributed systems, focusing on fault tolerance, scalability, and performance.

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Books Recommended:

Text books:

1. Andrew S. Tanenbaum and Maarten Van Steen, —Distributed Systems: Principles and Paradigms, 4th edition, Word Press 2023.
2. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems: Concepts and Design", 5th Edition, Pearson Education, 2012.

Reference Books:

1. Distributed Systems An Algorithmic Approach 2Ed By Sukumar Ghosh CRC press 2020.
2. Decentralized Systems and Distributed Computing by Sandhya Avasthi, Suman Lata Tripathi, Namrata Dhanda, and Satya Bhushan Verma Wiley Publication 2024.



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Web resources:

1. Distributed Computing: <https://www.coursera.org/courses/distributedcomputing>
2. Distributed Systems : <https://www.udemy.com/course/building-distributed-systems>

Online Courses: Nptel /Swayam Courses:

1. Distributed Systems by Rajiv Mishra IIT Karagpur
https://onlinecourses.nptel.ac.in/noc21_cs87

Evaluation Scheme:

Continuous Assessment (A)

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 Assignment / course project / group discussion /presentation / quiz/ any other for 10 marks.

Laboratory: (Term work)

Term work shall consist of minimum 8 experiments and a Mini project.

The distribution of marks for term work shall be as follows:

- i. Laboratory work (Performance of Experiments, Write-up): 15 Marks
- ii. Journal documentation (Write-up and/or Assignments): 10 marks

Semester End Examination (B):

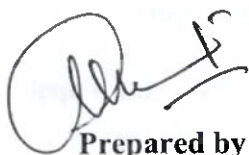
Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

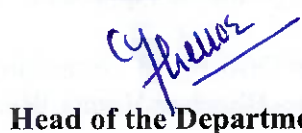
Laboratory:

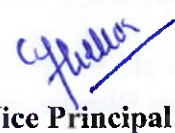
Oral examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work


Prepared by


Checked by


Head of the Department


Vice Principal


Principal



Program: B. Tech in Computer Science and Engineering (IoT and Cybersecurity with Block chain Technology)								Third Year B.Tech		Semester: V	
Course: Computer Vision								Course Code: DJS23BPE303			
Course: Computer Vision Laboratory								Course Code: DJS23BPE303L			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Assignm ent	Total	100
				60			15	15	10	40	
				Laboratory Examination			Term work				
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project presentation Assignment		50	
				25	--	--					15

Prerequisite:

1. Python Programming
2. Statistics for Engineers.

Course Objectives: The Objective of the course is

1. Understand the fundamental principles and concepts of computer vision systems.
2. Analyse visual data to identify patterns and extract meaningful information from images and videos.
3. Apply computational and learning-based techniques to solve real-world computer vision problems.
4. Design intelligent vision-based solutions for practical applications.

Course Outcomes: On successful completion of this course, students should be able to:

1. Explain the concepts of imaging geometry, camera models, and image formation in computer vision systems.
2. Apply edge detection techniques to identify important features in images.
3. Analyse feature extraction methods to represent and describe image characteristics.
4. Understand and apply deep learning techniques for visual data processing.
5. Analyse image classification and object detection methods for recognizing visual objects.
6. Analyse motion and optical flow techniques for understanding dynamic scenes in videos.



Unit	Description	Duration
1	Imaging Geometry and Camera Models: Fundamentals of image formation, Image acquisition process and imaging pipeline, Basic imaging geometry and coordinate systems, Pinhole camera model, Camera calibration, Perspective projection and orthographic projection Camera transformations: Translation and rotation, Rigid transformations Affine and projective transformations, Radiometry	7
2	Edge Detection Fundamentals of feature detection and image features, Types of features: edges, corners, blobs and regions Edge detection: Gradient-based edge detection- Sobel, Prewitt operators, Laplacian and Laplacian of Gaussian (LoG), Canny edge detector, Hough Transform.	6
3	Feature Extraction: Corner detection: Harris corner detector, Hessian Corner Detection, Difference of Gaussian (DoG), Feature descriptors: HOG, SIFT, SURF, GLOH	7
4	Deep Learning in Computer Vision: Introduction to deep learning in computer vision, Limitations of traditional computer vision methods, Basics of Neural Networks: Convolutional Neural Networks: Introduction to CNN, Convolution operation, Filters and feature maps, Stride and padding, Pooling layers, fully connected layers.	8
5	Image Classification and Object Detection: Image Classification Algorithms: Concept of image classification, Feature-based classification: Naïve Bayes, Support Vector Machine, Object Detection Algorithms: Concept of object detection, Difference between classification and localization, Sliding window technique, Region-based detection. Application of CNN for image classification and Object detection	7
6	Motion Analysis and Optical Flow: Introduction to motion in computer vision, Types of motion: rigid and non-rigid motion, Applications of motion analysis in computer vision. Basics of motion estimation, Temporal changes and frame difference, Optical flow concept and assumptions, Optical flow methods: Horn-Schunck method, Lucas-Kanade method. Motion detection techniques: Frame differencing, Background subtraction	7
Total		42

List of Laboratory Experiments:

Sr. No.	Suggested Experiments
1	Study image acquisition and perform basic image transformations (translation, rotation, scaling).
2	Implement geometric transformations such as affine and projective transformations on images.
3	Perform edge detection using Sobel, Prewitt, Laplacian, and Canny operators.
4	Detect lines using Hough Transform.
5	Implement corner detection using Harris Corner Detector



6	Extract features using descriptors such as HOG, SIFT, or SURF
7	Build a simple Convolutional Neural Network (CNN) for image classification.
8	Perform image classification using machine learning algorithms such as SVM or Naïve Bayes.
9	Implement object detection using sliding window or a pre-trained CNN-based model.
10	Implement Harris Corner Detector.
11	Compute optical flow using Lucas-Kanade or Horn-Schunck methods.
12	Mini Project (A group of 3 to 4 students is required to develop an application using Image processing and Computer Vision technology and submit report).

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Textbooks:

1. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Pearson Education, 4th Edition, 2021.
2. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer, New York, 2nd Edition, 2022.
3. David A. Forsyth and Jean Ponce, Computer Vision: A Modern Approach, Pearson Education, 2nd Edition, 2009.
4. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 1st Edition, 2016.
5. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow, O'Reilly Media, 3rd Edition, 2022.

Reference Books:

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2nd Edition, 2004.
2. Mark Nixon and Alberto Aguado, Feature Extraction and Image Processing for Computer Vision, Academic Press, 3rd Edition, 2019.
3. Adrian Kaehler and Gary Bradski, Learning OpenCV 4: Computer Vision with Python, O'Reilly Media, 1st Edition, 2017.
4. Simon J. D. Prince, Computer Vision: Models, Learning, and Inference, Cambridge University Press, 1st Edition, 2012.
5. E. R. Davies, Computer Vision: Principles, Algorithms, Applications, Learning, Academic Press, 5th Edition, 2018.

Web resources:

1. Edge Detection Using OpenCV
<https://learnopencv.com/edge-detection-using-opencv/>
2. Edge Detection in Computer Vision using OpenCV
<https://debuggercafe.com/edge-detection-in-computer-vision-using-opencv/>
3. Sobel vs Canny Edge Detection Techniques
<https://medium.com/@haidarlina4/sobel-vs-canny-edge-detection-techniques-step-by-step-implementation-11ae6103a56a>
4. Optical Flow Tutorial (MIT CBMM)
<https://cbmm.mit.edu/video/tutorial-computer-vision-4817>



Online Courses: NPTEL / Swayam:

1. Computer Vision By Prof. Jayanta Mukhopadhyay, IIT Kharagpur
https://onlinecourses.nptel.ac.in/noc25_cs143/preview
2. Deep Learning for Computer Vision, By Prof. Vineeth N Balasubramanian, IIT Hyderabad
https://onlinecourses.nptel.ac.in/noc21_cs93/preview
3. Computer Vision and Image Processing - Fundamentals and Applications By Prof. M. K. Bhuyan, IIT Guwahati.
https://onlinecourses.nptel.ac.in/noc23_ee39/preview

Evaluation Scheme:

Evaluation Scheme:

Semester End Examination (A):

Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

Oral examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.

Continuous Assessment (B):

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 Assignment / course project / group discussion / presentation / quiz/ any other for 10 marks.

Laboratory: (Term work)

1. Term Work shall consist of at least 6 practical based on the above list.
2. The distribution of marks for term work shall be as follows:
 - i. Laboratory work (Performance of Experiments, Write-up): 15Marks
 - ii. Mini Project/Case study/Presentation: 10 Marks

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work.

Prepared by

Checked by

Head of the Department

Vice Principal

Principal



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)								T.Y.B.Tech		Semester: V	
Course: Cognitive Computing								Course Code: DJS23BPE304			
Course: Cognitive Computing Laboratory								Course Code: DJS23BPE304L			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	100
				60			15	15	10	40	
				Laboratory Examination			Term work			Total Term work	
3	2	—	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project presentation Assignment			
				25	--	--	15	10	25		

Pre-requisite:

1. Python Programming

Objectives: The objective of the course is

1. To understand the key concepts and principles of Cognitive Computing.
2. To preview role of NLP's impact on Cognitive Computing Architectures
3. To apply training and testing procedures for language models for cognitive computing applications.
4. To integrate different cognitive computing methodologies.
5. To explore the full range of applications in Cognitive Computing.

Outcomes: On completion of the course, learner will be able to:

1. Understand the fundamentals of Cognitive Computing.
2. Explain the role of NLP in Cognitive Systems
3. Demonstrate understanding of techniques for text-based processing of natural language with respect to morphology.
4. Perform POS tagging for a given natural language and select a suitable language modelling technique based on the structure of the language
5. Verify the syntactic and semantic correctness of sentences using grammar and labelling.
6. Explore the diverse range of applications within cognitive computing.

**Detailed Syllabus: (unit wise)**

Unit	Description	Duration
1	Introduction to Cognitive Computing: Overview of Cognitive Computing, Cognitive Computing - Cognitive Psychology - The Architecture of the Mind, The Nature of Cognitive Psychology, Cognitive architecture, Cognitive processes, The Cognitive Modeling Paradigms, Declarative / Logic based Computational cognitive modeling – connectionist models, Bayesian models. Introduction to Knowledge-Based AI, Human Cognition on AI, Cognitive Architectures	06
2	Natural language processing in Cognitive systems: Natural Language Processing in support of a Cognitive System: Role of NLP in a cognitive system, semantic web, Applying Natural language technologies to Business problems. Representing knowledge in Taxonomies and Ontologies: Representing knowledge, Defining Taxonomies and Ontologies, knowledge representation, models for knowledge representation, implementation considerations.	07
3	Introduction to Natural Language Processing: Basic Knowledge and Grammar in language processing, Stages in NLP, Ambiguities and its types in English and Indian Regional Languages, Challenges of NLP, Applications of NLP. Word Level Analysis: Morphology Analysis –Survey of English Morphology, Inflectional Morphology & Derivational Morphology, Lemmatization, N-Grams, Unigrams/Bigrams Language Models, Corpora, Computing the Probability of Word Sequence, Training and Testing.	08
4	Syntax Analysis: Part-Of-Speech Tagging (POS) - Open and Closed Words. Tag Set for English (Penn Treebank), Rule Based POS Tagging, Transformation Based Tagging, Stochastic POS Tagging and Issues –Multiple Tags & Words, Unknown Words. Hidden Markov Model (HMM), Maximum Entropy, And Conditional Random Field (CRF). CFG: Derivations, Constituency, Phrase Structure and Dependency Structure	07
5	Semantic Analysis and Pragmatics: Lexical Semantics, Attachment for Fragment of English- Sentences, Noun Phrases, Verb Phrases, Prepositional Phrases, Relations Among Lexemes & Their Senses – Homonymy, Polysemy, Synonymy, Hyponymy, WordNet, Robust Word Sense Disambiguation (WSD), Lexical Disambiguation, Resolving Lexical Ambiguity, Lexical Ambiguity Resolution Pragmatics: Discourse –Reference Resolution, Reference Phenomenon, Syntactic & Semantic Constraints on Co Reference	08
6	Application of cognitive computing: Health care application: Foundations of cognitive computing for healthcare, constituents in healthcare ecosystem and learning from patterns in healthcare data. Banking & Fraud Prevention: analyzing transaction patterns, payment fraud, identity theft and money laundering. Transportation & Logistics: Dynamic Route Optimization for traffic and weather. Media and Entertainment: Cognitive systems recommend tailored content based on user preferences and behavior.	06
Total		42



List of Laboratory Experiments:	
Sr. No.	Suggested Experiments
1	Preprocessing steps in NLP Chunking using NLTK and SPACY
2	Apply various other text preprocessing techniques for any given text : Stop Word Removal, Lemmatization / Stemming.
3	Perform morphological analysis and word generation for any given text.
4	Implement N-Gram model for the given text input.
5	Build a POS tagger using HMM
6	Compare the accuracy of rule-based POS tagging, stochastic POS tagging, and transformation-based tagging. Use a common dataset for evaluation.
7	Compare the effectiveness of syntactic and semantic constraints on reference resolution in a pragmatic context. Evaluate their contribution to resolving reference phenomena.
8	Implement TF-IDF vectors in Natural Language Processing
9	Generate recursive set of sentences using Context Free Grammar Identify the word senses using "synset" in NLTK
10	Implement similarity detection in NLP
11	Implement Named Entity Recognizer for the given text input.
12	Students are supposed to complete any one case study not limited to following list of topics. a) Sequence Prediction b) Chatbot Using Deep Learning c) Traffic Sign Classification d) Automatic Music Generation e) Music Genre Classification f) Text Summarizer



Books Recommended:

Text Books

1. Hurwitz, Kaufman, and Bowles, Cognitive Computing and Big Data Analytics, Wiley, First edition, 2015
- 2 Masood, Adnan, Hashmi, Adnan, "Cognitive Computing Recipes-Artificial, Intelligence Solutions Using Microsoft Cognitive Services and TensorFlow, 2015
3. Speech and Language Processing, 2nd Edition, Jurafsky and Martin, Prentice Hall, 2000, ISBN: 0130950696

Reference Books

1. Peter Fingar, Cognitive Computing: A Brief Guide for Game Changers, PHI Publication, 2015
2. Gerardus Blokdyk, Cognitive Computing Complete Self-Assessment Guide, 2018
3. Rob High, Tanmay Bakshi, Cognitive Computing with IBM Watson: Build smart applications using Artificial Intelligence as a service, IBM Book Series, 2019
4. Ayyadevara V K., Reddy Y, Modern Computer Vision with PyTorch: Explore deep learning concepts and implement over 50 realworld image applications, Pakt Publishing, Kindle edition available, 2020.
5. Manning C., Schutze H. (latest reprint). Foundations of Statistical Natural Language Processing, The MIT Press, Kindle edition available. 1999
6. James Allen. Natural Language Understanding. The Benajmins/Cummings Publishing Company Inc. . ISBN 0-8053-0334-0. 1994
7. Hagiwara M. Real-World Natural Language Processing: Practical applications with deep learning, Manning Publications. 2021
8. Kamath U., Liu J., Whitaker J, Deep Learning for NLP and Speech Recognition, Springer, Kindle edition available, 2019.

Web resources

1. Natural Language processing

<https://www.coursera.org/specializations/natural-language-processing>

2. Google AI

<https://ai.google/get-started/for-developers/>

3. Python & NLTK by Udemy

<https://www.udemy.com/course/nlp-natural-language-processing-with-python/>



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Online Courses: NPTEL / Swayam

1. Natural Language Processing, By Prof. Pawan Goyal, IIT Kharagpur,
https://onlinecourses.nptel.ac.in/noc24_cs39/preview
2. Natural Language Processing with Deep Learning in Python
<https://www.udemy.com/course/natural-language-processing-with-deep-learning-in-python/>

Evaluation Scheme:

Continuous Assessment (A)

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 Assignment / course project / group discussion / presentation / quiz/ any other for 10 marks.

Laboratory: (Term work)

Term work shall consist of minimum 8 experiments and a Mini project.

The distribution of marks for term work shall be as follows:

- i. Laboratory work (Performance of Experiments, Write-up): 15 Marks
- ii. Journal documentation (Write-up and/or Assignments): 10 marks

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work.

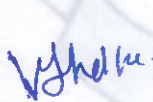
Semester End Examination (B):

Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

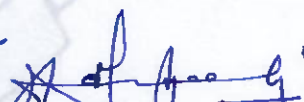
Oral examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.


Prepared by


Checked by


Head of the Department


Vice Principal


Principal



Shri Vile Parle Kelavani Mandal's

DWARKADAS J.SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)						T.Y.B.Tech		Semester : V		
Course: Artificial Intelligence						Course Code: DJS23BMD301				
Course: Artificial Intelligence Laboratory						Course Code: DJS23BMD301L				
Teaching Scheme (Hours / week)				Evaluation Scheme						
				Semester End Examination Marks (A)		Continuous Assessment Marks (B)			Total marks (A+ B)	
Lectures	Practical	Tutorial	Total Credits	Theory		Term Test 1	Term Test 2	Term Test 3	Total	100
				60		15	15	10	40	
				Laboratory Examination		Term work				
3	2	--	3	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation / Assignment	Total Term work	50
				--	--	25	15	10	25	

Pre-requisite:

1. Basic programming languages
2. Data Structures

Course Objectives: The objective of the course is:

1. To create a thorough understanding of AI basics and real-time applications in its sub-domains.
2. To explore AI techniques like informed, uninformed and adversarial searching to solve real-life problems in a state space tree representation.
3. To explore various approaches to knowledge representation, reasoning, and planning strategies in AI systems.
4. To understand the scope of Generative Networks in AI.



Course Outcomes: On completion of the course, learners will be able to:

1. Develop a basic understanding of AI building blocks presented in intelligent agents.
2. Design an appropriate problem-solving method for an agent to find a sequence of actions to reach the goal state.
3. Analyze various AI approaches to knowledge– intensive problem solving, reasoning and planning.
4. Understand the concept of planning and analyze different types of planning strategies.
5. Acquire basic knowledge of Expert System.
6. Analyze the working and architecture for Generative Networks.

Detailed Syllabus:

Unit	Description	Duration
1	Introduction to Artificial Intelligence: Introduction, History of Artificial Intelligence, Intelligent Systems: categorization of Intelligent System, Components of AI Program, Foundations of AI, Sub-areas of AI, Current trends in AI Intelligent Agents: Agents and Environments, The concept of rationality, The nature of environment, The structure of Agents, Types of Agents, Learning Agent.	07
2	Problem solving: Solving Problem by Searching: Problem Solving Agent, Formulating Problems, Example Problems. Uninformed Search Methods: Breadth First Search (BFS), Depth First Search (DFS), Depth Limited Search, Depth First Iterative Deepening (DFID) Informed Search Methods: Greedy Best First Search, A* Search Stochastic Local Search Algorithms and Optimization Problems: Hill climbing search, Simulated Annealing, Genetic algorithms, Ant Colony Optimization. Adversarial Search: Game Theory, Algorithm Minimax, Alpha-Beta Pruning.	12
3	Knowledge and Reasoning: Knowledge based Agents, The WUMPUS World, Inference in FOL, Forward chaining, Backward chaining, Knowledge Engineering in First- Order Logic, Unification, Resolution.	08
4	Planning: The planning problem, Planning with State Space Search, STRIPS, Goal Stack Planning, Planning graphs, Partial order planning, Hierarchical Planning.	06
5	Expert System: Introduction, Phases in building Expert Systems, ES Architecture, Case Study on MYCIN Rule based system.	04
6	Generative AI: Introduction to Gen AI, Overview of generative models and their importance in AI, Fundamentals of Probability theory and generative modeling, Introduction to GANs, Generative Adversarial Network algorithm, VAEs and other generative models. Significance of generative models, Challenges with generative models.	05
Total		42



List of Laboratory Experiments:	
Sr.No.	Suggested Experiments
1	Select a problem statement relevant to AI. Identify the problem PEAS Description Problem formulation
2	To implement BFS/DFS/DFID search algorithms to reach goal state.
3	To implement A* search algorithm to reach goal state.
4	To implement Local Search algorithm: World Block Problem using Hill climbing search
5	To illustrate Game playing.
6	To implement Wumpus world AI Problem.
7	To implement alpha beta pruning.
8	To implement Tic-tac-toe AI Problem.
9	To implement 8-Queens Problem AI Problem.
10	Case study on Planning Problem. Identify and analyze a planning problem
11	To implement Generative Adversarial Network (GAN) algorithm.
12	Case Study on analysis of an AI Application from IEEE, Science Direct, and Scopus-Indexed Journal papers.
13	Mini Project (A mini project to be undertaken by a group of 4–5 students, focusing on topics related to the subject.)

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Books Recommended:**Text Books**

1. Stuart J. Russell and Peter Norvig, Artificial Intelligence A Modern Approach ,Fourth Edition" Pearson Education,2022.
2. Saroj Kaushik , Artificial Intelligence, Cengage Learning, 1st Edition, 2011.
3. George F Luger, Artificial Intelligence, Pearson Education., 6th Edition,2021.
4. Foster, D., Generative deep learning, O'Reilly Media, Inc. 2022
5. Koller, D. and Friedman, N., Probabilistic graphical models: principles and techniques. MIT press, 2009.

Reference Books

1. Elaine Rich and Kevin Knight ,Artificial Intelligence, Third Edition,2017.
2. Patrick Henry Winston, Artificial Intelligence, Addison-Wesley, Third Edition.1992
3. Lavika Goel , Artificial Intelligence concept and applications, WILEY Publishers, 2021
4. N.P.Padhy , Artificial Intelligence and Intelligent Systems, Oxford University Press. 2005.
5. Dr. Nilakshi Jain., Artificial Intelligence, WILEY Publishers, First Edition,2019.

Web resources:

1. Microsoft AI School- <https://www.microsoft.com/en-us/ai>
2. Google AI Education- <https://ai.google/why-ai/>
3. Practical tutorials and courses <https://docs.fast.ai/>



Online Courses: NPTEL / Swayam

1. Course on- Fundamentals Of Artificial Intelligence- By Prof. Shyamanta M. Hazarika, IIT Guwahati
https://onlinecourses.nptel.ac.in/noc23_ge40/preview
2. Course on - Artificial Intelligence: Search Methods for problem Solving- By Prof. Deepak Khemani, IIT Madras https://onlinecourses.nptel.ac.in/noc23_cs92/preview

Evaluation Scheme:

Continuous Assessment (A)

Theory:

1. Term Test 1 (based on 40 % syllabus) of 15 marks for the duration of 45 min.
2. Term Test 2 (on next 40 % syllabus) of 15 marks for the duration of 45 min.
3. Term Test 3 can be conducted as Assignment / course project / group discussion / presentation / quiz/case study for 10 marks.

Laboratory: (Term work)

Term work shall consist of minimum 8 experiments.

The distribution of marks for term work shall be as follows:

- i. Laboratory work (Performance of Experiments): 10 Marks
- ii. Assignments: 5 Marks
- ii. Mini project (Implementation, Report): 10 marks

The final certification and acceptance of term work will be subject to satisfactory performance of laboratory work, and upon fulfilling minimum passing criteria in the term work.

Semester End Examination (B):

Theory:

1. Question paper will be based on the entire syllabus summing up to 60 marks.
2. Total duration allotted for writing the paper is 2 hrs.

Laboratory:

Oral and Practical examination will be based on the entire syllabus including, the practical's performed during laboratory sessions.

Prepared by

Checked by

Head of the Department

Vice Principal

Principal



Shri Vile Parle Kelavani Mandal's

DWARKADAS J.SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)								T.Y.B.Tech		Semester : V	
Course: Environmental Studies								Course Code: DJS23XHS331T			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	50
				--			--	--	--	--	
				Laboratory Examination			Term work				
--	--	1	1	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Assignment		Total Term work	
				--	--	--	--	25		25	

Prerequisite: -

1. Interest in Environment and its impact on Humans.

Course Objectives: The objective of the course is:

1. Familiarise students with environment related issues such as depleting resources, pollution, ecological problems and the renewable energy scenario.
2. Give overview of Green Technology options.

Course Outcomes: On completion of the course, learner will be able to:

1. Understand how human activities affect environment.
2. Understand the various technology options that can make a difference.



Unit	Detailed Syllabus:	Duration
1	Air Pollution Sources of Air pollution. Definition of Air Quality Index and how it is measured.	2
2	Water Pollution Sources of water pollution. Ground water pollution and eutrophication.	2
3	Noise Pollution Noise pollution and sources. Decibel limits for hospital, library, silence zone.	1
4	Biodiversity loss Value of Biodiversity. Endangered species.	1
5	Deforestation Product and services provided by forests. Relationship between forests and climate change.	2
6	Renewable Energy sources Our energy needs and global energy crisis. Renewable energy sources.	2
7	Climate change Greenhouse gases and climate change.	2
8	Green Technology Data Center Energy Efficiency, Thin-Client and Energy Efficiency.	2
	Total	14

Websites:**CITES:** www.cites.org**Convention on Biological Diversity:** www.biodiv.org**Kalpvriksh:** www.kalpvriksh.org**Water pollution:** http://en.wikipedia.org/wiki/Water_pollution**Ecosan:** www.eco-solutions.org



Text books:

1. Environmental Studies from Crisis to Cure, R. Rajagopalan, Oxford University Press, Second Edition,
2. Textbook of Environmental Studies for Undergraduate Courses, Erach Bharucha for University Grants Commission, New Delhi & Bharti Vidyapeeth Institute of Environment Education and Research.
3. Environmental Pollution: Principles, Analysis and Control; P. Narayanan, CBS Publishers
4. Green Information Technology: A Sustainable Approach, Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, Morgan and Kaufman, Elsevier

Reference Books:

1. Information Technologies in Environmental Engineering: New Trends and Challenges, Paulina Golinska, Marek Fortsch, Jorge Marx-Gómez, Springer, 2011.

List of Tutorials

SN.	Tutorial List
1	Case study on Smog.
2	Qualitative and Quantitative methods for Air Pollution Monitoring
3	Presentation on Water Pollution (Industrial, Sewage) explaining any specific case.
4	General Techniques in Water Quality Monitoring
5	List effects of noise pollution on human health. Measure decibel level in college library, canteen, classroom
6	Case study on effect of Pollution on Biodiversity loss.
7	Radioactive and Hazardous Pollutants.
8	Debate for and against "To promote Economic growth, Deforestation is required."
9	Presentation on different Renewable Energy Technologies.
10	Green Hydrogen: The Energy of Future
11	Report on major impact of Global warming on Environment giving real examples.
12	Report on advantages and examples of Green Building for Sustainable development,



13	Sustainable Software Design: Reducing energy/carbon footprint (e.g., green coding, optimized AI/ML, efficient cloud usage)
14	Reducing AI's carbon footprint (efficient models, serverless) and applying AI for environmental good, like optimizing energy grids, smart buildings, precision agriculture (water/fertilizer use), waste management, and climate modeling.

Tutorial: (Term work: 25 marks)

- Minimum of 10 Tutorials required.
- The distribution of marks for term work shall be as follows:

- i. Performance in Tutorial: 15 Marks
- ii. Write-up/ Report: 10 marks

The final certification and acceptance of term work will be subject to satisfactory work performance and fulfilling minimum passing criteria in the term work.

Prepared by

Checked by

Head of the Department

Vice Principal

Principal



Program: B.Tech in Computer Science and Engineering(IoT and Cybersecurity with Block chain Technology)								T.Y.B.Tech		Semester : V	
Course : Innovative Product Development III								Course Code: : JS23XSC301P			
Teaching Scheme (Hours / week)				Evaluation Scheme							
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				Total marks (A+ B)
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	
				--			--	--	--	--	
				Laboratory Examination			Term work			Total Term work	
--	2	--	1	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project presentation Assignment			
				--	--	25	-	--	25		

Objectives: The objective of the course is.

1. To acquaint the students with the process of identifying the need (considering a societal requirement) and ensuring that a solution is found out to address the same by designing and developing an innovative product.
2. To familiarize the students with the process of designing and developing a product, while they work as part of a team.
3. To acquaint the students with the process of applying basic engineering fundamentals, so as to attempt at the design and development of a successful value added product.
4. To inculcate the basic concepts of entrepreneurship and the process of self-learning and research required to conceptualize and create a successful product.

Outcome: On completion of the course, learner will be able to:

1. Identify the requirement for a product based on societal/research needs.
2. Apply knowledge and skills required to solve a societal need by conceptualizing a product, especially while working in a team.
3. Use standard norms of engineering concepts/practices in the design and development of an innovative product.
4. Draw proper inferences through theoretical/ experimental/simulations and analyse the impact of the proposed method of design and development of the product.



5. Develop interpersonal skills, while working as a member of the team or as the leader.
6. Demonstrate capabilities of self-learning as part of the team, leading to life-long learning, which could eventually prepare themselves to be successful entrepreneurs.
7. Demonstrate product/project management principles during the design and development work and also excel in written (Technical paper preparation) as well as oral communication.

Guidelines for the proposed product design and development:

- Students shall form a team of 3 to 4 students (max allowed: 5-6 in extraordinary cases, subject to the approval of the department review committee and the Head of the department).
- Students should carry out a survey and identify the need, which shall be converted into conceptualization of a product, in consultation with the faculty supervisor/head of department/internal committee of faculty members.
- Students in the team shall understand the effective need for product development and accordingly select the best possible design in consultation with the faculty supervisor.
- Students shall convert the best design solution into a working model, using various components drawn from their domain as well as related interdisciplinary areas.
- Faculty supervisor may provide inputs to students during the entire span of the activity, spread over 2 semesters, wherein the main focus shall be on self-learning.
- A record in the form of an activity log-book is to be prepared by each team, wherein the team can record weekly progress of work. The guide/supervisor should verify the recorded notes/comments and approve the same on a weekly basis.
- The design solution is to be validated with proper justification and the report is to be compiled in a standard format and submitted to the department. Efforts are to be made by the students to try and publish a technical paper, either in the institute journal, "Techno Focus: Journal for Budding Engineers" or at a suitable publication, approved by the department research committee/ Head of the department.
- The focus should be on self-learning, capability to design and innovate new products as well as on developing the ability to address societal problems. Advancement of entrepreneurial capabilities and quality development of the students through the year long course should ensure that the design and development of a product of appropriate level and quality is carried out, spread over two semesters, i.e. during the semesters V and VI.

Guidelines for Assessment of the work:

- The review/ progress monitoring committee shall be constituted by the Head of the Department. The progress of design and development of the product is to be evaluated on a continuous basis, holding a minimum of two reviews in each semester.
- In the continuous assessment, focus shall also be on each individual student's contribution to the team activity, their understanding and involvement as well as responses to the questions being raised at all points in time.
- Distribution of term work marks during semester V shall be as given below:
 - Marks awarded by the supervisor based on performance 10
 - Marks awarded by Review Committee 10
 - Quality of Documentation 05



Review/progress monitoring committee may consider the following points during the assessment.

- In the semester V, the entire design proposal shall be ready, including components/system selection as well as the cost analysis.
- Two reviews will be conducted based on the presentation given by the student's team.

Tentative rubrics for reviews will be as follows:

Review 1:

- i. Implementation Details & Status (60% project implementation)
- ii. Design & System Specifications
- iii. Presentation Quality
- iv. Contribution as a team member and Punctuality
- v. Project Documentation.

Review 2:

- i. Implementation Details & Status (90% project implementation)
- ii. Draft copy of research paper , draft copy of Copy right or Patent if applicable .

First review is based on readiness of building the working prototype.

Second review shall be based on a presentation as well as the demonstration of the working model. This review will also look at the readiness of the proposed technical paper presentation of the team.

The overall work done by the team shall be assessed based on the following criteria:

1. Quality of survey/ need identification of the product.
2. Clarity of Problem definition (design and development) based on need.
3. Innovativeness in the proposed design.
4. Feasibility of the proposed design and selection of the best solution.
5. Cost effectiveness of the product.
6. Societal impact of the product.
7. Functioning of the working model as per stated requirements.
8. Effective use of standard engineering norms.
9. Contribution of each individual as a member or the team leader.
10. Clarity on the write-up and the technical paper prepared.

Guidelines for Assessment of Semester Reviews:

- The write-up should be prepared as per the guidelines given by the department.
- The design and the development of the product shall be assessed through a presentation and demonstration of the working model by the student team to a panel of Internal and



Shri Vile Parle Kelavani Mandal's

DWARKADAS J.SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



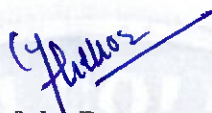
External Examiners, preferably from industry or any research organisations having an experience of more than five years, approved by the Head of the Institution. Students are compulsorily required to present the outline of the technical paper prepared by them during the final review in semester VI.


Dyesh

Prepared by


H. S. J.

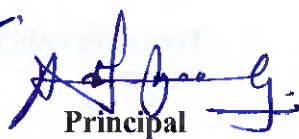
Checked by


Y. K. J.

Head of the Department


Y. K. J.

Vice Principal


A. B. J.

Principal





Shri Vile Parle Kelavani Mandal's

DWARKADAS J.SANGHVI COLLEGE OF ENGINEERING

(Empowered Autonomous College Affiliated to the University of Mumbai)



Program: First Year Computer Science and Engineering (IoT and Cybersecurity with Block chain Technology)								Semester: V			
Course: Artificial Intelligence for Cybersecurity								Course Code: DJS23BSC301			
Teaching Scheme (Hours / week)				Evaluation Scheme							Total marks (A+ B)
				Semester End Examination Marks (A)			Continuous Assessment Marks (B)				
Lectures	Practical	Tutorial	Total Credits	Theory			Term Test 1	Term Test 2	Term Test 3	Total	
				--			15	15	10	40	40
				Laboratory Examination			Term work				
1	-	-	1	Oral	Practical	Oral & Practical	Laborato ry Work	Tutorial / Mini project / presentation/ Journal			-
				--	--	--	-	--	--	--	

Coursera Course: Artificial Intelligence for Cybersecurity

Link: [Artificial Intelligence for Cybersecurity | Coursera](#)

Number of hours: 25 Hrs, Intermediate level

About the Course:

Artificial intelligence (AI) is revolutionizing how organizations safeguard digital assets, detect threats, and respond to cyberattacks. This course provides a deep understanding of how AI can be leveraged to enhance cybersecurity, enabling professionals to build intelligent systems that predict and prevent potential breaches.

Learners will explore how AI-driven techniques streamline security operations, identify anomalies, and improve decision-making in real-time. By the end of the course, you'll be able to design AI-based solutions that strengthen defense mechanisms and address modern cybersecurity challenges. What makes this course unique is its focus on practical, hands-on implementation of AI tools and algorithms in security workflows. It bridges the gap between theory and practice, providing both conceptual clarity and real-world case studies. This course is ideal for cybersecurity professionals, machine learning practitioners, and students interested in combining AI with security. A basic understanding of Python and machine learning concepts is recommended. Based on the book, Artificial Intelligence for Cybersecurity, by Bojan Kolosnjaji, Huang Xiao, Peng Xu, and Apostolis Zarras.



Course Outcomes: On successful completion of this course, student should be able to understand

1. Recognize AI as a powerful tool for intelligence analysis in cybersecurity.
2. Design AI-based solutions for cybersecurity challenges.
3. Explore the components and workflow of AI security solutions.

Detailed Syllabus:		
Unit	Description	Duration
1	Big data in cybersecurity Big data's role in cybersecurity, focusing on threat detection, incident response, and ethical considerations, automation in cybersecurity, AI's role in cybersecurity	3
2	AI Problems and Methods AI, ML, and statistics, and explore ML taxonomies, limitations, and security risks, random forest, K-means, and GANs for cybersecurity applications.	3
3	Malware detection and network intrusion analysis AI-driven malware detection and network intrusion analysis, focusing on dataset utilization, model implementation, and real-world threat classification, UEBA techniques for detecting advanced threats using AI-driven anomaly detection and numerical feature extraction fraud, phishing, and spam detection	3
4	Threat Intelligence Threat intelligence retrieval and AI applications for analyzing cyber threats, anomaly detection techniques for industrial control systems, enhancing security through practical methods and frameworks, threat detection, vulnerability discovery, and secure workflow design.	3
5	Data quality role in AI and LLMs validation, cleaning, correlation, causation, bias, and variance in AI for cybersecurity, evaluating AI models using metrics, monitoring performance for latency and bias, and implementing human-in-the-loop strategies for continuous improvement in cybersecurity.	3
6	AI security challenges Adversarial machine learning (AML) concepts, vulnerabilities in generative AI, and defensive techniques, AI security challenges, privacy, accountability, and trust, summary	4
Total		25

Continuous Assessment (B)

Evaluation Scheme:

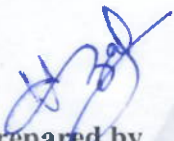
1. The identified courses shall focus on developing AI-related competencies and skills among students. Each selected course shall carry one (1) credit.
2. Faculty members shall act as mentors for the Coursera courses identified for various semesters of the program.
3. One faculty mentor shall be assigned to each division of the program for effective coordination and monitoring of this activity.



4. The assigned faculty mentor shall complete and review the course content before assigning the course to students.
5. The term work for these one-credit courses shall form a part of the Internal Continuous Assessment (ICA) component.
6. The Term work (ICA) component shall carry **40 marks**

The distribution of marks for term work shall be as follows:

- **15 Marks:** Coursera Course Completion and Certification
- **15 Marks:** Faculty-curated quiz conducted through the Coursera/Any other platform
- **10 Marks:** Internal faculty evaluation through quizzes, assignments, presentations, or Project


Prepared by


Checked by


Head of the Department


Vice Principal


Principal