Shri Vile Parle Kelavani Mandal's

Dwarkadas J. Sanghvi College of Engineering

(Empowered Autonomous College Affiliated to the University of Mumbai)

Course Structure and Syllabus

of

M.Tech.

in

Computer Engineering

Prepared by:- Board of Studies in Computer Engineering

Recommended by:- Academic Council of D. J. Sanghvi College of Engineering

Approved by:- Governing Body of D. J. Sanghvi College of Engineering

Revision: 3 (D26)

With effect from the Academic Year: 2026-2027

Shri Vile Parle Kelavani Mandal's
DWARKADAS J.SANGHVI COLLEGE OF ENGINEERING
(Empowered Autonomous College Affiliated to the University of Mumbai)

Scheme for First Year M. Tech Program in Computer Engineering: Semester I (Autonomous)
(Academic Year 2026-2027)

Sr	Course Code	Course	Teaching Scheme				End Semester Examination						Continuous Assessment			Aggregate (A+B)	Credits earned
			Theory (hrs.)	Practical (hrs.)	Tutorial (hrs.)	Credits	Duration (Hrs)	Theory	Oral	Pract	Oral & Pract	ESE Total (A)	Term Test	Term Work	CA Total (B)		
1	D26CPC501	Advanced Algorithm and Complexity	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26CPC501L	Advanced Algorithm and Complexity Laboratory	--	2	--	1	--	--	25	--	--	25	--	25	25	50	1
2	D26CPC502	Big Data Infrastructure	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26CPC502L	Big Data Infrastructure Laboratory	--	2	--	1	--	--	25	--	--	25	--	25	25	50	1
3@	D26CPE501	Deep Learning	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26CPE501L	Deep Learning Laboratory	--	2	--	1	--	--	25	--	--	25	--	25	25	50	1
	D26CPE502	Advanced System Security and Digital Forensics	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26CPE502L	Advanced System Security and Digital Forensics Laboratory	--	2	--	1	--	--	25	--	--	25	--	25	25	50	1
	D26CPE503	Data Engineering	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26CPE503L	Data Engineering Laboratory	--	2	--	1	--	--	25	--	--	25	--	25	25	50	1
	D26CPE504	Generative AI and Prompt Engineering	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26CPE504L	Generative AI and Prompt Engineering Laboratory	--	2	--	1	--	--	25	--	--	25	--	25	25	50	1
4	D26CSC501P	Mini Project-I	--	4	--	2	--	--	50	--	--	50	--	50	50	100	2
5#	D26XOE511	Data Analytics	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26XOE512	Intellectual Property & Patenting	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26XOE513	Cyber Security and Laws	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26XOE514	Agile Frameworks	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26XOE515	Design of Experiments	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
	D26XOE516	Operations Research	3	--	--	3	2	60	--	--	--	60	40	--	40	100	3
Total			**12**	**10**	**--**	**17**	**--**	**240**	**125**	**--**	**--**	**365**	**160**	**125**	**285**	**650**	**17**

@ Any 1 Department Level Elective
# Any 1 Institue Level Elective

Prepared by **Checked by** **Head of Dept** **Vice Principal** **Principal**

Continuous Assessment

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 / Tutorial	Performance in the laboratory, tutorial & / assignment. (ICA Component Evaluation).	25	

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

Semester End Assessment (B):

Course	Assessment Tools	Marks	Time (hrs.)
Theory	Written paper based on the entire syllabus.	60	2
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

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering	Semester : I
Course: Advanced Algorithm and Complexity	Course Code: D26CPC501
Course: Advanced Algorithm and Complexity Laboratory	Course Code: D26CPC501L

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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	50
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				25	--	--	15	10		25	

Pre-requisite: Knowledge of

1. Data structures
2. Analysis of Algorithms
3. Discrete structures and Set Theory

Objectives:

1. To analyze the algorithms using space and time complexity.
2. To acquire knowledge of various applied algorithms.
3. To understand algorithms that have applications in areas such as geometric modelling, graphics, robotics, vision, computer animation, etc.

Outcomes:

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

1. Analyze the correctness and running time of the algorithms that are implemented in several domains.
2. Apply the algorithms and design techniques to formulate the optimized solution.
3. Understand and apply various advanced data structures to solve computing problems.
4. Introduce and practice advanced algorithms and programming techniques necessary for developing sophisticated computer application programs

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Fundamentals of Algorithms:** Introduction to Algorithms, analyzing algorithms (Big Oh, small Oh, Omega, small Omega, Theta, Tilde), Growth of Functions, Recurrences, the substitution method, the recursion-tree method, the master method, Complexity of Recursive algorithms, Proving Techniques. Amortized Analysis (Aggregate, Accounting and Potential Method)	06
2	**Dynamic Programming and Linear Programming:** Elements of dynamic programming, Matrix-chain multiplication, Weighted Job Scheduling Algorithm, Graphical Method, Simplex Method, Standard and slack forms, Formulating problems as linear programs.	08
3	**Advanced Data Structures:** Top-Down Splay Tree, Skew Heaps, Fibonacci Heaps, van Emde Boas Trees, AA-Trees, Treaps, Data Structures for Disjoint Sets.	06
4	**Graph Algorithms:** Applications of DFS- Undirected Graphs, Biconnectivity, Euler circuits, Directed Graphs, Cyclic Graphs: Hamiltonian and Eulerian Cycles, Single-Source Shortest Paths-The Bellman-Ford algorithm, All-Pairs Shortest Paths-The Floyd-Warshall algorithm.	08
5	**Streaming Algorithms:** Basic Definitions, Sampling, Sketching, Counting distinct Items, Heavy Hitters Problem, CountSketch Algorithm.	06
6	**Advanced Algorithms:** Multithreaded Algorithms, String Matching - The naive string-matching algorithm, The Rabin-Karp algorithm, String matching with finite automata, The Knuth-Morris Pratt algorithm, Number-Theoretic Algorithms- Elementary number-theoretic notions, Greatest common divisor, Modular arithmetic, Solving modular linear equations, The Chinese remainder theorem, Powers of an element, Randomized Algorithms-Monte Corlo& Las Vegas Algorithms	08
	Total	42

List of Laboratory Experiments:

1. Implement various sorting algorithms and analyze them w.r.t. Time and Space complexity.
2. Implement Matrix chain multiplication.
3. Implement Weighted Job scheduling.
4. Implement Linear Programming Simplex method.
5. Implement Disjoint Sets (Union-Find) and apply them to Kruskal's MST.
6. Implement self-adjusting search trees — Splay Tree and Treap.
7. Implement DFS applications — articulation points/biconnectivity, Euler circuits, and topological sort.
8. Implement Single-Source Shortest Paths-The Bellman-Ford algorithm.
9. Implement All-Pairs Shortest Paths-The Floyd-Warshall algorithm.
10. Implement streaming algorithms — Count-Min/Count-Sketch heavy hitters and distinct-item counting.
11. Implement Naïve String matching and Rabin Karp algorithm.

12. Implement the Knuth-Morris-Pratt algorithm and string matching with finite automata.
13. Implement number-theoretic and randomized algorithms (GCD, modular exponentiation, CRT, Miller-Rabin, Las Vegas).
14. Study research paper on Advanced Data Structures and Streaming Algorithms.

Books Recommended:

Text books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", PHI, India Second Edition, February 2010.
2. H. S. Wilf, "Algorithms and complexity", Prentice hall, Summer 1994.
3. Horowitz, Sahani and Rajsekaran, Fundamentals of Computer Algorithms", Galgotia, January 2008.

Reference Books:

1. The Design and Analysis of Computer Algorithms" by Aho, Hopcroft, Ullman, 1974.
2. "Algorithm Design" by Kleinberg and Tardos, 16 March 2005.
3. "Data Structures and Algorithm Analysis in C" by Mark Allen Weiss, Pearson, September 1996.

Web Resources:

1. https://onlinecourses.nptel.ac.in/e-learning/preview/noc25_cs150
2. https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs67

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering							Semester : I				
Course: Big Data Infrastructure							Course Code: D26CPC502				
Course: Big Data Infrastructure Laboratory							Course Code: D26CPC502L				
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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	50
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				25	--	--	15	10		25	

Pre-requisite: Knowledge of Databases, Java, Python

Objectives:

1. To introduce students to current scenarios and various facets of big data and also to create an awareness on the concepts of cloud computing and virtualization.
2. To equip them with necessary knowledge to use the tools for solving various big data problems in different domains
3. To enable students to have skills that will help them to solve complex real-world problems in for decision support.

Outcomes:

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

1. Develop problem solving and critical thinking skills in fundamental enabling techniques like Hadoop and Mapreduce in big data analytics.
2. Develop Data management capabilities for large scale data processing by using various bigdata technologies and APIs.
3. Evaluate Data at Scale-Working with Big Data
4. Analyze statistical data using various analytical tools.
5. Visualise data using various tools

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction to Big Data and Hadoop Infrastructure** Big data definition, enterprise / structured data, social / unstructured data, unstructured data needs for analytics, what is Big Data? **Introduction of Big data programming** Hadoop, History of Hadoop, The ecosystem and stack, The Hadoop Distributed File System (HDFS), Components of Hadoop, Design of HDFS, Java interfaces to HDFS, Architecture overview, Development **Environment, Hadoop distribution and basic commands.** Developing a Map Reduce Application, How Map Reduce Works, The MapReduce Anatomy of a Map Reduce Job run, Failures, Job Scheduling, Shuffle and Sort, Task execution, Map Reduce Types and Formats, Map Reduce Features, Real-World MapReduce.	08
2	**NoSQL Databases:** **NoSQL Vs SQL** - Structured and Unstructured Data, Taxonomy and NoSQL Implementation, NoSQL Architectural Patterns, Using NoSQL to manage BigData, **MongoDb** - Basic architecture of MongoDb Types of NoSql Databases, Searching and Indexing Big Data. **NoSQL Case Studies**- Google's BigTable, Mongo DB, Neo4J, Amazon DynamoDB	06
3	**Programming with Hive/Pig:** Data warehouse system for Hadoop, Optimizing with Combiners and Partitioners, Bucketing, More common algorithms: sorting, indexing and searching, Relational manipulation: map-side and reduce-side joins, evolution, purpose and use, Engine for executing data flows in parallel on Hadoop: Overview, comparison and architecture, Latin scripting and statements, data types, UDF's, built in functions and use cases.	06
4	**Introduction to Apache Spark and Use Cases** **Apache Spark APIs for large-scale data processing**: Overview, Linking with Spark, Initializing Spark, Resilient Distributed Datasets (RDDs), External Datasets, RDD Operations, Passing Functions to Spark, Working with Key-Value Pairs, Shuffle operations, RDD Persistence, Removing Data, Shared Variables, deploying to a Cluster Spark Streaming, Spark MLlib and ML APIs, Spark Data Frames/Spark SQL, Integration of Spark and Kafka, Map reduce, Mongodb with spark	08
5	**Data Streams and Streaming Analytics** Data streams and stream analytics. Spark architecture and components. Popular Spark platforms, DataBricks. Spark programming and tools, SparkML library for Machine Learning.	06
6	**Dashboard Creation and Visual Story Telling:** Dashboard Design principles, Effective Dashboard Display Media, Dashboard creation using visualization tools for the use cases: Finance- marketing-insurance-healthcare etc.,	06
	Total	**42**

List of Laboratory Experiments:

1. Installing Hadoop Framework on a Linux based Platform
2. Implement Map reduce program to perform sorting of numbers
3. Install MongoDB and perform CRUD operations
4. Install Neo4j and perform CRUD operations
5. Performing Hive Commands
6. Implementing PIG scripting for data handling
7. Perform Map reduce using Apache Spark
8. Configure a public cloud
9. Perform Data Stream Analytics
10. Perform data visualisation using tools like Gephi, Google API etc.

Books Recommended:

Text Book:

1. Rinku Sharma Dixit and Shailee Lohmor Choudhary, "Big Data Analytics: Using Artificial Intelligence Technologies Transforming Organizations", IK International Publishing House, 2021.
2. Spark: The Definitive Guide: Big Data Processing Made Simple by Bill Chambers, Matei Zaharia, O'Reilly, Feb 2018
3. Enterprise NoSQL for Dummies, Mark Logic Special Edition. Wiley, Nov 2017
4. Practical Hive: A Guide to Hadoop's Data Warehouse System by Scott Shaw, Andreas François Vermeulen, Ankur Gupta, David Kjerrumgaard, Apress, August 2016
5. Big Data for Dummies, By Judith S. Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufma, Wiley,1st Edition, 2013
6. "Mastering Cloud Computing", Rajkumar Buyya, Christian Vecchiolla and S Thamara Selvi, Tata Mc Graw Hill publication, 2013
7. "Cloud Computing Bible", Barrie Sosinsky, Wiley publication, Jan 2011

Reference Book:

1. Apache Spark Quick Start Guide: By Shrey Mehrotra, Akash Grade, Packt Publishing Ltd., 2019
2. Handbook of Big Data Technologies, edited by Albert Y. Zomaya, Sherif Sakr,Springer, 2017
3. Mining of Massive Datasets by Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, Cambridge University Press ,2010
4. Hadoop Application Architectures: Designing Real-World Big Data Applications by Mark Grover, Ted Malaska, Jonathan Seidman, Gwen Shapira, O'Reily, 2015
5. A Handbook of Statistical Analyses Using R, By Torsten Hothorn, Brian S. Everitt, CRC Press, 2006
6. OpenStack Operations, David Stilson, O'Reilly, 2017.

Web Resources:

1. https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_cs65
2. https://www.coursera.org/specializations/big-data

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering										Semester : I
Course: Deep Learning										Course Code: D26CPE501
Course: Deep Learning Laboratory										Course Code: D26CPE501L
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	Assignment	Total
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	50
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal		
				25	--	--	15	10		25	

Pre-requisite: Knowledge of

1. Artificial Intelligence
2. Machine Learning

Objectives:

1. To understand the Hyper Parameter Tuning.
2. To explore Deep Learning Techniques with different learning strategies.
3. To design Deep Learning Models for real time applications.

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

1. Understand the fundamentals and tools of Deep Learning.
2. Apply Hyperparameter Tuning techniques.
3. Design and analyze Convolutional and Sequential Models.
4. Develop and evaluate Generative and Adversarial Networks.
5. Investigate suitable Deep Learning models for real-world applications.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction to Deep Learning:** Overview of Neural Network, Deep learning and human brain, Why is Deep Learning taking off?, Deep Learning applications Overview of Tools: Torch, TensorFlow, Keras	04
2	**Convolutional Neural Network:** Introduction to CNNs, Kernel filter, Principles behind CNNs, Multiple Filters, CNN applications **ConvNet Architectures** Discussions on famous convnet architectures: AlexNet, VGG, GoogLeNet, ResNet	09
3	**Hyperparameter Tuning, Batch Normalization** Tuning Process, Using an Appropriate Scale to pick Hyperparameters, Hyperparameters Tuning in Practice: Pandas vs. Caviar, Normalizing Activations in a Network, Fitting Batch Norm into a Neural Network, why does Batch Norm work, Batch Norm at Test Time	06
4	**Sequential models:** Introduction to Sequence Models and RNNs, Recurrent Neural Network Model, Backpropagation Through Time, Different Types of RNNs: Unfolded RNNs, Seq2Seq RNNs, Long Short-Term Memory (LSTM), Bidirectional RNN, Vanishing Gradients with RNNs, Gated Recurrent Unit (GRU), RNN applications, Sequence models and attention mechanism, Attention over Images, Hierarchical Attention	09
5	**Adversarial Networks** Introduction to adversarial Networks, Auto encoders (standard, denoising, contractive, etc.), Variational Auto encoders, Generative Adversarial Networks, Diffusion Models: Basics of Diffusion models, Advantages of over GANs, Popular Diffusion tools: Stable Diffusion, DALL-E, Midjourney, Applications of Adversarial Networks	09
6	**Deep Learning Case Studies:** Image Processing, Natural Language Processing, Speech Recognition, Video Analytics	05
	Total	42

List of Laboratory Experiments:

1. To understand and implement Artificial Neural Networks (ANN) using PyTorch/TensorFlow for classification tasks.
2. To implement image classification using CNN.
3. To study and compare AlexNet, VGG16, and ResNet architectures using transfer learning.
4. To perform hyperparameter tuning of a deep neural network and analyze its impact on model performance.
5. To implement Recurrent Neural Networks (RNN) for sequence prediction using PyTorch/TensorFlow
6. To implement Long Short-Term Memory (LSTM) networks for sentiment analysis or text classification.
7. To visualize Attention Mechanisms in sequence models using PyTorch/TensorFlow.
8. To perform image caption generation using CNN-LSTM architectures.
9. To Implement Autoencoders for Image Reconstruction.
10. To Generate Handwritten Digits using GANs
11. To generate images using pretrained Diffusion Models (Stable Diffusion/DALL·E) and analyze prompt engineering techniques.
12. Study of Explainable AI Tools: SHAP and LIME for Deep Learning Model Interpretation

Books Recommended:

Text books:

1. Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016.
2. Umberto Michelucci, Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection, 2019.
3. Michael Nielsen (Goodreads Author), "Neural Networks and Deep Learning",2015
4. TensorFlow 1.x Deep Learning Cookbook, Gulli and Kapoor, Packt, 2017.

Reference Books:

1. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
2. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.
3. Neural Networks: A Systematic Introduction, Raúl Rojas, 1996.
4. David Foster, Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play O"Relly, 2019.
5. Maxim Lapan, Deep Reinforcement Learning HandsOn: Apply modern RL methods, with deep Q- networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more, Packt 2018.
6. Santanu Pattanaya K, Pro Deep Learning with TensorFlow A Mathematical Approach to Advanced Artificial Intelligence in Python, APress, 2017.

Online Resources:

1. NPTEL:

Deep Learning, By Prof. Prabir Kumar Biswas, IIT Kharagpur
https://onlinecourses.nptel.ac.in/noc22_cs22/preview

2. Coursera:

Deep Learning Specialization, By DeepLearning.AI
https://www.coursera.org/specializations/deep-learning#courses

Program: First Year M. Tech. Computer Engineering										Semester : I			
Course: Advanced System Security and Digital Forensics										Course Code: D26CPE502			
Course: Advanced System Security and Digital Forensics Laboratory										Course Code: D26CPE502L			
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	Assignment	Total			
				60			15	15	10	40		100	
				Laboratory Examination			Term work			Total Term work		50	
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal					
				25	--	--	15	10		25			

Pre-requisite: Knowledge of

1. Computer Networks
2. Databases
3. Operating Systems

Objectives:

1. To understand cyber-attacks and defense strategies.
2. To understand underlying principles of access control mechanisms.
3. To explore software vulnerabilities, attacks, and protection mechanisms of wireless networks and protocols, mobile devices, and web applications.
4. To develop and mitigate security management and policies.
5. To understand and explore techniques used in digital forensics.

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

1. Understand cyber-attacks, apply access control policies, and control mechanisms.
2. Identify malicious code and targeted malicious code.
3. Detect and counter threats to web applications.
4. Understand the vulnerabilities of Wi-Fi networks and explore different measures to secure wireless protocols, WLAN and VPN networks.
5. Understand the ethical and legal issues associated with cybercrimes and be able to mitigate impact of crimes with suitable policies.
6. Use different forensic tools to acquire and duplicate data from compromised systems and analyse the same.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction & Access Control:** **Introduction:** Need for Cyber Security, what is Cybercrime, Types of Cybercrime, Threats vs Vulnerabilities, Defense Strategies and Techniques, Authentication Methods, and Protocols **Access Control Policies:** DAC, MAC, Rule Based Access Control, Role Based Access Control, Multi-level Security Models: Biba Model, Bell La Padula Model, Single Sign on, Federated Identity Management.	08
2	**OS Security:** Separation, Memory and Address Protection: Fencing, Relocation, Base/Bound Registers, Segmentation and Paging. Linux and Windows Vulnerabilities, File System Security	06
3	**Web Application Security:** OWASP Top 10 Web Vulnerabilities, Cookies and their role in Cyber Attacks, SSL/TLS, HTTP vs HTTPS, SSH, Privacy on the Web, Web Browser Attacks, Account Harvesting, Web Bugs, Clickjacking, Cross Site Scripting(XSS) vs Cross-Site Request Forgery (CSRF), Session Hijacking , Phishing vs Vishing vs SMShing vs Pharming, Web Service Security: SOAP Protocol, OAuth 2.0	08
4	**Digital Forensics and Incident Response:** **Digital Forensics:** Introduction to Digital Forensics, The Need for Digital Forensics, Types of Digital Forensics. **Incident and Initial Response:** Introduction to Computer Security Incident, Goals of Incident response, Incident Response Methodology, Initial Response, Formulating Response Strategy.	06
5	**Forensic Duplication and Analysis** **Forensic Duplication:** Introduction to Forensic Duplication, Types of Forensic Duplicates, Introduction to Forensic Duplication Tools. **Data Acquisition:** Introduction to Static and Live/Volatile Data, Static Data Acquisition from Windows (FTK Imager), Static Data Acquisition from Linux (dd/dcfldd), Live Data Acquisition from Windows (FTK Imager) Forensic Investigation and Analysis: Forensic Analysis of acquired data in Linux/Windows, Investigating Logs and Registry files.	08
6	**Evidence Handling and Forensic Reporting** **Evidence Handling:** Digital Evidence: Types and characteristics, Challenges for Evidence Handling, Evidence Handling Methodology, Chain of Custody **Forensic Reporting:** Goals of a Report, Layout of an Investigative Report, Guidelines for writing a report, Sample Forensic Report	06
	Total	**42**

List of Laboratory Experiments:

1. Static Code Analysis using open-source tools. Recommended Tool: Flawfinder Python Distribution
2. Web Application Vulnerability Scanning and Auditing using open-source tools. Recommended Tools: Nikto / Wapiti / Burpsuite
3. Study and exploit database flaws and vulnerabilities using SQL Injection Attack. Recommended Tool: SQLMap
4. Study and Implement Packet Sniffing using Open-Source Tools. Recommended Tools: Wireshark, TCP Dump
5. Study and implement Session Hijacking / Man in the Middle (MiTM) attack in a controlled virtual environment. Recommended Tools: Ettercap / Bettercap
6. Penetration Testing and Vulnerability Exploitation. Recommended Tool: Metasploit (Kali Linux)
7. Exploring Router and VLAN security, setting up access lists. Recommended Tool: Cisco Packet Tracer (Student Edition)
8. Static and Live Data Acquisition from Windows Recommended Tool: FTK Imager , TCP Dump
9. Static Data Acquisition from Linux Recommended Tool: dd, dcfldd
10. Analysis of Forensic Duplicates (Recommended Tool: Autopsy)

Books Recommended:

Text books:

1. Computer Security Principles and Practice, William Stallings, Fourth Edition, Pearson Education, 2019
2. Cryptography and Network Security – Principles and Practice, William Stallings, Seventh Edition, Pearson Education, 2017
4. Security in Computing, Charles P. Pfleeger, Fifth Edition, Pearson Education, 2018
5. Network Security and Cryptography, Bernard Menezes, Cengage Learning, 2010
6. Network Security Bible, Eric Cole, Second Edition, Wiley, 2009

Reference books:

1. Incident Response & Computer Forensics by Kevin Mandia, Chris Prosise, Wiley, 2nd Edition. 2014
2. Computer Security, Dieter Gollman, Third Edition, Wiley,3rd Edition, 2011
3. Digital Forensic by Nilakshi Jain & Kalbande, Wiley, 2016
4. Cyber Security. Nina Godbole, Sunit Belapure, Wiley, 2011
5. Build your own Security Lab, Michael Gregg, Wiley India, 2012
6. CCNA Security, Study Guide, Tim Boyles, Sybex, 2010
7. Web Application Hacker's Handbook, Dafydd Stuttard, Marcus Pinto, Wiley India, 2008
8. Network Infrastructure Security, Randy Waver, Dawn Weaver, Cengage Learning, 2009

Web Resources (For our Reference):

1. http://www.opentechinfo.com/learn-use-kali-linux/
2. https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Prepared by Checked by Head of the Department Principal

<table>
<tr><td colspan="8">Program: First Year M. Tech. Computer Engineering</td><td colspan="5">Semester : I</td></tr>
<tr><td colspan="8">Course: Data Engineering</td><td colspan="5">Course Code: D26CPE503</td></tr>
<tr><td colspan="8">Course: Data Engineering Laboratory</td><td colspan="5">Course Code: D26CPE503L</td></tr>
<tr><td colspan="4" rowspan="2">Teaching Scheme
(Hours / week)</td><td colspan="9">Evaluation Scheme</td></tr>
<tr><td colspan="3">Semester End Examination Marks (A)</td><td colspan="5">Continuous Assessment Marks (B)</td><td rowspan="2">Total marks (A+ B)</td></tr>
<tr><td rowspan="3">Lectures</td><td rowspan="3">Practical</td><td rowspan="3">Tutorial</td><td rowspan="3">Total Credits</td><td colspan="3">Theory</td><td>Term Test 1</td><td>Term Test 2</td><td>Assignment</td><td>Total</td></tr>
<tr><td colspan="3">60</td><td>15</td><td>15</td><td>10</td><td>40</td><td>100</td></tr>
<tr><td colspan="3">Laboratory Examination</td><td colspan="3">Term work</td><td rowspan="2">Total Term work</td><td rowspan="3">50</td></tr>
<tr><td>3</td><td>2</td><td>--</td><td>4</td><td>Oral</td><td>Practical</td><td>Oral & Practical</td><td>Laboratory Work</td><td colspan="2">Tutorial / Mini project / presentation/ Journal</td></tr>
<tr><td></td><td></td><td></td><td></td><td>25</td><td>--</td><td>--</td><td>15</td><td colspan="2">10</td><td>25</td></tr>
</table>

Pre-requisite: Database systems, SQL.

Objectives:

1. Understand the role of data engineering in modern computing and explain the complete data engineering lifecycle from initialization to serving across batch and stream paradigms.
2. Develop proficiency in data storage technologies including RDBMS, NoSQL databases, distributed file systems, data lakes, and warehouses.
3. Design, build, and optimize scalable data pipelines using ETL/ELT, workflow orchestration, real-time processing, and governance frameworks while incorporating emerging trends like lakehouse, DataOps, and data mesh.

Outcomes:

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

1. Design end-to-end data architectures for structured, semi-structured, and unstructured data using appropriate batch and stream processing models.
2. Implement and optimize data ingestion, transformation, and storage solutions using modern tools for both real-time and batch analytics, ensuring data quality and lineage.
3. Evaluate and apply data governance, security, cost optimization, and emerging architectural patterns like lakehouse and data mesh to enterprise-scale data systems.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction to Data Engineering:** Role of Data Engineering in modern computing, Data Engineering lifecycle, Structured, semi-structured, and unstructured data, Batch processing vs stream processing, Data ecosystems and architectures, Introduction to data-driven systems.	6
2	**Data Storage and Database Systems:** Relational Database Systems (RDBMS), NoSQL databases: Key-Value stores, Document databases, Column-family databases, and Graph databases, Distributed file systems, Data lakes and data warehouses, CAP theorem and distributed storage concepts.	6
3	**Data Ingestion and ETL Pipelines:** Data ingestion techniques, ETL and ELT architectures, Data extraction and transformation methods, Data integration strategies, Workflow scheduling and orchestration, Pipeline automation concepts.	8
4	**Real-Time Data Engineering:** Event-driven architectures, Stream processing concepts, Message queues and event brokers, Real-time analytics pipelines, Windowing and stream aggregation, Applications of IoT and monitoring systems.	8
5	**Data Governance and Optimization:** Data Governance: Data quality dimensions, Data validation and cleansing, Metadata management, Data lineage and cataloging, Data governance frameworks, Privacy, compliance, and security mechanisms, Data Optimization: Query optimization techniques, Data partitioning and indexing, Caching and memory optimization, Scalability and fault tolerance, Performance monitoring and tuning, Cost optimization strategies.	8
6	**Emerging Trends in Data Engineering:** DataOps principles, Lakehouse architecture, AI-assisted data engineering, Data mesh concepts, Integration of data engineering with MLOps, Future trends in large-scale data systems	6
	Total	42

List of Laboratory Experiments:

1. RDBMS Lab. Load dataset into PostgreSQL. Execute database operations & measure latency.
2. NoSQL Lab. Load dataset into MongoDB. Execute database operations & measure latency.
3. Explore basic python libraries for Data Engineering.
4. Exploring Data Types with Pandas. Handle structured, semi-structured, unstructured data. Load CSV, JSON, and .txt log file. Show schema, parse JSON nesting, extract text entities.
5. RDBMS vs NoSQL Comparison Lab. Compare query patterns on structured data. Discuss CAP implications.
6. Schema Design for Data Warehouse. Practice dimensional modeling.
   Design star schema. Create dim and fact tables, run OLAP queries.
7. Batch ETL with Python. Build basic ETL pipeline.
   Extract data from API + CSV, clean nulls/outliers, transform with joins, load to SQLite / Parquet in local MinIO/S3/ etc.
8. Incremental Data Ingestion with CDC. Understand change data capture.
   Set up Debezium + Postgres in Docker. Update source table, capture changes to Kafka topic.
9. Workflow Orchestration using Apache Airflow. Automate pipeline execution.
   Create DAG with 3 tasks: extract API, transform, load to warehouse. Add retry + SLA.
10. Stream Processing with Kafka + Python. Build real-time pipeline.

Produce mock IoT sensor data to Kafka. Consume, compute 1-min rolling average, print alerts.

11. Windowing & Aggregation in Spark Structured Streaming. Practice event-time processing.
    Read Kafka stream in PySpark. Apply tumbling and sliding windows to count events per device.
12. Data Lakehouse Table Formats. Compare lakehouse features.
    Create Delta Lake and Iceberg tables. Test schema evolution, time travel, and partition pruning.
13. Data Quality & Testing with Great Expectations. Implement data validation.
    Define expectations for a dataset: no nulls in PK, values in range. Run checks, generate data docs.
14. Metadata & Lineage with OpenLineage + DataHub. Track data lineage end-to-end.
    Run Airflow DAG integrated with OpenLineage. View job/column-level lineage in DataHub UI.

Books Recommended:

Text books:

1. Reis, J. & Housley, M. Fundamentals of Data Engineering, O'Reilly, 2022.
2. Akidau, T., Chernyak, S., & Lax, R. Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing, O'Reilly, 2018.

Reference Books:

1. Kleppmann, M. Designing Data-Intensive Applications, O'Reilly, 2017
2. Dehghani, Z. Data Mesh: Delivering Data-Driven Value at Scale, O'Reilly, 2022

Web Resources (For our Reference):

1. https://www.tutorialspoint.com/data-engineering/index.htm
2. https://www.geeksforgeeks.org/data-engineering/data-engineering-tools-and-skills/
3. https://www.w3schools.com/sql/default.asp

Program: First Year M. Tech. Computer Engineering	Semester : I
Course: Generative AI and Prompt Engineering	Course Code: D26CPE504
Course: Generative AI and Prompt Engineering Laboratory	Course Code: D26CPE504L

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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	50
3	2	--	4	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				25	--	--	15	10		25	

Pre-requisite: Artificial Intelligence, Data Mining, Introduction to Machine Learning

Objectives:

1. To equip students with the skills to understand and apply generative AI models and prompt engineering techniques to create and optimize AI-generated content across diverse applications.

Outcomes:

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

1. Analyze LLM architectures, training methods, and ethics
2. Apply prompt engineering techniques
3. Evaluate models, architectures, and fine-tuning approaches
4. Design advanced AI applications and agentic systems
5. Build and deploy end-to-end Generative AI solutions

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Foundations of Generative AI and Large Language Models** Taxonomy of Generative Models: VAEs, GANs, diffusion models, autoregressive models; Large Language Models: architecture overview (attention, transformer blocks); Pre-training, fine-tuning, and Instruction Tuning (InstructGPT, RLHF); Model families and capabilities: GPT-4, Claude 3, Gemini, LLaMA, Mistral; Expanded Ethics: prompt injection, deepfakes, EU AI Act, bias auditing, responsible deployment	07

2	**Prompt Engineering: Principles, Structure, and Parameter Control** The Five Principles of Prompting: Give Direction / Specify Format / Provide Examples / Evaluate Quality / Divide Labor; Anatomy of an effective prompt: system prompt, context, instruction, examples, output format; Prompt templates and structured prompt design for reproducibility; LLM Sampling Parameters: temperature, top-k, top-p, frequency penalty, presence penalty; Role prompting and persona assignment; Constraint handling, ambiguity management, and error handling in prompts; Output formatting: JSON, XML, Markdown, tables — structured output design; Automatic Prompt Engineering (APE): using AI to generate better prompts	08
3	**Generative Model Architectures and Fine-Tuning Strategies** Transformer Architecture deep-dive: self-attention, multi-head attention, positional encoding; Encoder-only (BERT), Decoder-only (GPT), Encoder-Decoder (T5, BART) models — when to use each; Fine-tuning vs. Prompting: task fit analysis, data requirements, cost trade-offs; Parameter-Efficient Fine-Tuning (PEFT): LoRA, Prefix Tuning, Adapters; Instruction Tuning and RLHF: how models learn to follow instructions; Evaluation metrics for generative models: perplexity, FID, BLEU, ROUGE	07
4	**Advanced Prompting Techniques** Zero-Shot, One-Shot, and Few-Shot prompting — construction, example selection guidelines; Chain-of-Thought (CoT): Zero-shot CoT, Few-shot CoT, Self-Consistency prompting; Tree-of-Thought (ToT): branch generation, evaluation, pruning, recommendation; Hallucination Control: grounding instructions, fallback clauses, factual anchoring; Prompt Chaining: output-as-input pipelines for complex multi-step tasks; Multi-modal Prompting: text + image inputs (GPT-4V, Gemini 1.5 Pro); Function Calling and Structured Output: JSON schema enforcement for tool integration; Prompt Security: prompt injection, jailbreaking, indirect injection, defensive strategies	07
5	**Prompt Evaluation, Optimization and Responsible AI** Prompt Evaluation Metrics: BLEU, ROUGE-L, BERTScore, human evaluation protocols; A/B Prompt Testing: designing controlled experiments to compare prompt variants; Automated Prompt Optimisation: DSPy, APE, LLM-as-evaluator frameworks; Bias Auditing in Prompts: detecting and mitigating systematic bias in outputs; Content moderation and safety in deployed systems	07
6	**Generative AI Applications, Governance and Emerging Trends** Domain Applications — with evaluation case studies: • Healthcare: clinical note summarisation, differential diagnosis prompting • Legal: contract analysis, regulatory compliance checking • Software Engineering: code generation, review, debugging prompts • Education: personalised tutoring systems, assessment generation Regulatory Landscape: EU AI Act, data protection considerations, IP and copyright; Research Frontiers: Constitutional AI, self-critique prompting, world models	06

List of Laboratory Experiments:

1. Exploring LLM Parameter Sensitivity: Understand how sampling parameters (temperature, top-k, top-p, max_tokens) affect the creativity, coherence, and reliability of LLM outputs.
2. Prompt Design and Evaluation using ROUGE & BERTScore: Design summarisation prompts of increasing specificity and measure the quality improvement using automated evaluation metrics.

3. Chain-of-Thought vs. Direct Prompting — Reasoning Accuracy Study: Empirically demonstrate the accuracy improvement of Chain-of-Thought prompting over direct prompting on multi-step reasoning tasks.
4. Few-Shot Prompt Construction for Multi-class Text Classification: Build a robust Few-Shot text classifier and investigate the impact of example quality and quantity on classification accuracy.
5. Multi-turn Conversational Agent with Memory: Build a stateful conversational assistant using LangChain's memory modules and compare coherence with and without conversation history.
6. Hallucination Detection and Grounded Question Answering: Design and test grounding strategies that prevent LLM hallucination in document-based Q&A, & quantify the hallucination rate with & without grounding.
7. Prompt Chaining Pipeline for Multi-Step Content Generation: Build a multi-step prompt chaining pipeline using LangChain SequentialChain, demonstrating how complex tasks can be decomposed into verifiable sub-steps.
8. Mini Project: Design, build, evaluate, and present an intelligent domain-specific. The system must demonstrate measurable improvement over a naive LLM baseline.

Books Recommended:

Text books:

1. James Phoenix, Mike Taylor, Prompt Engineering for Generative AI, O'Reilly Media, Inc., 2024
2. David Foster, Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play, O'Relly Media, 2019
3. Rafael Valle, Hands-On Generative Adversarial Networks with Keras: Build Advanced GAN Architectures Using Keras and TensorFlow", Packt Publishing, 2019
4. Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016.
5. Rajalingappaa Shanmugamani, Deep Learning for Computer Vision" Packt Publishing, 2018

Reference Books:

1. Navveen Balani, Prompt Engineering: Unlocking Generative AI: Ethical Creative AI for All", Amazon Digital Services LLC – Kdp, 2023
2. Stuart J. Russell and Peter Norvig, "Artificial Intelligence A Modern Approach" 3rd Edition, Pearson Education 2010
3. Daniel Jurafsky, James H. Martin—Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, Pearson Publication, 2014.
4. Alexander Clark (Editor), Chris Fox (Editor), Shalom Lappin (Editor) "The Handbook of Computational Linguistics and Natural Language Processing", July 2010

Web Resources (For our Reference):

1. NPTEL: Prompt Engineering https://onlinecourses.swayam2.ac.in/e-learning/preview/imb26_mg131
2. NPTEL: Fundamentals of Generative AI https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs95
3. Microsoft Learn: Generative AI and Prompt Engineering: https://learn.microsoft.com/training/paths/introduction-generative-ai/?utm_source=chatgpt.com
4. https://www.tutorialspoint.com/prompt_engineering/index.htm
5. https://www.promptingguide.ai/

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering							Semester : I				
Course: Mini Project-I							Course Code: D26CSC501P				
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	Assignment	Total	
				--			--	--	--	--	--
				Laboratory Examination			Term work			Total Term work	100
--	4	--	2	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				50	--	--	--	50		50	

Pre-requisite:

1. Domain knowledge of any Program Specific Outcome (PSO) of the Computer Engineering curriculum.

Objectives:

1. Project Implementation with reference to the subjects in the various domains of Computer Engineering, UG & PG curriculum, with a view to strengthen research.
2. To explore and identify real-world social and industrial problems, to realize feasible solutions with added business value, based on conventional or innovative methods/practices.

Outcomes:

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

1. Identify problem statement, design and develop project in predefined timeline.
2. Provide problem solutions by learning/exploring various ideas from multi-disciplinary domains across different disciplines.
3. Draw proper inferences through theoretical/ simulations/ experimental and analyze the impact of the proposed method towards design and development of the product.
4. Develop and enhance skills associated with literature survey, hardware and software co-integrations, documentation, development and testing leading to innovative product or a paper publication in a reputed journal.

Syllabus:

Domain knowledge (any field of knowledge and beyond) needed from the following areas for the effective implementation of the product:

Algorithms, Data structures, Networking and Internet of Things, Data science and Big Data, Robotics, Artificial Intelligence (AI), Machine learning (ML), Image Processing, Security, Natural Language Processing.

The above areas can be updated (expanded), based on the needs of technological innovations and development needed for a specific project/product.

Evaluation scheme:

Every student will be reviewed individually once in a semester by review panel based on the following criteria:

1. Innovative ideas and Motivation
2. Objectives, Expected outcome and long-term social impact
3. Literature survey and Comparative Methodology
4. Timeline, progress and execution (Project Implementation)
5. Documentation/ synopsis of project report.
6. Overall presentation

Marks scored in the mid semester review will be considered as part of term work.

The final certification and acceptance of Term work ensure satisfactory performance and the outcome of evaluation centered about evaluation scheme.

Prepared by	Checked by	Head of the Department	Principal

Program: First Year M. Tech. Computer Engineering							Semester : I				
Course: Data Analytics							Course Code: D26XOE511				
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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	--
3	--	--	3	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				--	--	--	--	--		--	

Pre-requisite: Knowledge of

1. Fundamentals of probability
2. Applied Mathematics

Objectives:

To build the strong foundation in statistics which can be applied to analyze data and make predictions

Outcomes:

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

1. Interpret data using descriptive statistics
2. Demonstrate sampling distributions and estimate statistical parameters
3. Develop hypothesis based on data and perform testing using various statistical techniques.
2. Perform analysis of variance on data
3. Examine relations between data

Detailed Syllabus: (unit wise)

Unit	Description	Duration
1	**Introduction to Statistics** Types of statistics, population vs sample Measures of Central Tendency: arithmetic mean, properties, weighted mean, properties, median, mode, grouped and ungrouped data, empirical relation between the mean, median and mode, geometric mean, harmonic mean, relation between arithmetic, geometric and harmonic mean, outlier. Measures of dispersion: range, quartile deviation, mean deviation, standard deviation, properties, variance, root mean square deviation, empirical relations	06

	between measures of dispersion, absolute and relative dispersion, coefficient of variation, moments, Pearson's β and γ coefficients, skewness, kurtosis, population parameters and sample statistics, histogram, frequency polygon Measures of position: quartiles, interquartile range, semi interquartile range, percentiles, percentile rank, 10–90 percentile range, box and whisker plot	
2	**Sampling distribution and Estimation** Sampling distribution: Central limit theorem, population distribution, chi-square distribution, Z - distribution, student's t-distribution, F-Distribution. Statistical Estimation: Characteristics of estimators, consistency, unbiasedness, unbiased estimates, efficient estimates, sufficient estimators, point estimates, interval estimates, determination of sample size for estimating mean and proportions, estimates of population parameters, probable error	08
3	**Hypothesis Testing for data driven decision making** Hypothesis testing: Test of significance, null and alternative hypothesis, type I and type II error, factors affecting Type II error, probability of Type II error, power of test, p-Value, critical region, level of significance Confidence interval: Population mean, difference between two population means, population proportion, difference between two population proportions, variance, ratio of variances of two populations Goodness of fit test using Kolmogorov-Smirnov test and Anderson Darling test Tests using z-statistics: difference between sample proportion and population proportion, difference between two sample proportion, difference between sample mean and population mean with known σ and unknown σ, difference between two sample means, one tailed and two tailed tests Test using t-statistics: difference between sample mean and population mean, difference between two independent sample means, difference between means from the same group; Test using F-statistics: equality of population variance Test using chi-square statistics: test of independence, goodness of fit	12
4	**Analysis of Variance (ANOVA) for data analysis** Sample size calculation, one way ANOVA, POST-HOC Analysis (Tukey's Test), randomized block design, two way ANOVA	08
5	**Examining Relationship** Correlation: Scatter plot, covariance, Karl Pearson's coefficient of correlation, hypothesis test for correlation, correlation vs causation, extreme data values, limits of correlation coefficient, Rank correlation, Spearman's rank correlation coefficient, Repeated ranks, partial and multi correlation. Regression: linear regression analysis, lines of regression, regression coefficients, scatter plot with regression lines, hypothesis test for regression, multiple regression, coefficient of determination, residuals, collinearity, influential observations	08
	Total	42

Books Recommended:

Text Books:

1. Ken Black, *Business Statistics for Contemporary Decision Making*, John Wiley & Sons, Inc. Sixth Edition.
2. Anderson Sweeney Williams, *Statistics for Business and Economics*, Cengage Learning, 2011.

Reference Books:

1. Jay L. Devore, *Probability and Statistics for Engineering and the Sciences*, Cengage Learning, 2011.
2. Douglas C. Montgomery, George C. Runger, *Applied Statistics & Probability for Engineering*, John Wiley & Sons, Inc, 2002

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering							Semester : I				
Course: Intellectual Property & Patenting							Course Code: D26XOE512				
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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	--
3	--	--	3	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				--	--	--	--	--		--	

Objectives:

1. Understanding, defining and differentiating different types of intellectual properties (IPs)
2. Assessing different IP management (IPM) approaches
3. Exposure to the Legal management of IP and understanding of real life practice of IPM.

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

1. Recognize the crucial role of IP for the purposes of product and technology development.
2. Understand how and when to file a patent
3. Apply the knowledge to understand the entire ecosystem
4. Derive value from IP and leverage its value in new product and service development

Detailed Syllabus: (unit wise)

Unit No.	Topics	Duration
1	**Intellectual Property Law** Introduction and the need for intellectual property right (IPR), Intellectual Property laws, IPR in India: Genesis and development, Major International Instruments concerning Intellectual Property Rights: Paris Convention, the Berne Convention, the Universal Copyright Convention, the WIPO Convention, the Patent Cooperation Treaty, the TRIPS Agreement, Types of IPR	05
2	**Patents and Trademarks** Elements of Patentability: Novelty, Non Obviousness, Industrial Application, Non Patentable Subject Matter, Registration Procedure, Rights and Duties of Patentee, Assignment and licence, Restoration of lapsed Patents, Surrender and Revocation of Patents,	08

	Infringement, Remedies & Penalties, Patent office and Appellate Board, Case study of existing patents related to software, healthcare, devices. Concept of Trademarks, Different kinds (brand names, logos, signatures, symbols, well known marks, certification marks and service marks), Non Registrable Trademarks, Registration of Trademarks, Rights of holder and assignment and licensing of marks, Infringement, Remedies & Penalties, Trademarks registry and appellate board.	
3	**Copyrights and Design** Copyrights: Nature, Subject matter: original literary, dramatic, musical, artistic works, cinematograph films and sound recordings, Registration Procedure, Term of protection, Ownership of copyright, Assignment and licence of copyright, Infringement, Remedies & Penalties, Related Rights, distinction between related rights and copyrights Design: meaning and concept of novel and original, procedure for registration, effect of registration and term of protection	**10**
4	**Patenting** Introduction to the Indian Patent System , Patent Law as Concepts, IPR as a group of rights, Patent Rights, Fundamental of Patents, Patent Law in India, Understanding the Patents Act and the Rules.	**08**
5	**Patent Drafting and Searching** Anatomy of a patent application, Adequate disclosure, The art of drafting patent claims Patent searching: A. Purposes and techniques B. Available On-line tools	**06**
6	**Actions for patent infringement** Interpretation of claims, Doctrine of equivalents, Product testing as a possibly infringing use Doctrine of exhaustion, Legal and equitable remedies for infringement	**05**
		42

Books Recommended:

Text Books:

1. Feroz Ali, *The Law of Patents -With A Special Focus On Pharmaceuticals In India*, LexisNexis, 2011.
2. Ronald D. Slusky, *Invention Analysis and Claiming – A Patent Lawyer's Guide*, Second Edition, American Bar Association, 2012.
3. Feroz Ali, *The Touchstone Effect – The Impact of Pre-grant Opposition on Patents*, LexisNexis, 2009.

Reference Books:

1. Drucker. F. Peter*, Innovation and Entrepreneurship*, Harper business, 2006.
2. Deborah. E. Bouchoux*, Intellectual Property Rights,* Cengage Learning, 2013.
3. Prabuddha Ganguli, *Intellectual Property Rights– Unleashing The Knowledge Economy*, Tate Mc Graw Hill Publishing Company Ltd. 2001.
4. Martin Roger, *The Design of Business*, Harvard Business Publishing, 2009.

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering							Semester : I				
Course: Cyber Security and Laws							Course Code: D26XOE513				
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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	--
3	--	--	3	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				--	--	--	--	--		--	

Pre-requisite: Knowledge of

1. Computer Network
2. Information Security Fundamentals
3. Basic Understanding of Internet Technologies

Objectives:

1. To understand the fundamentals of cyber security, cybercrime, cyber threats, and cyber offences.
2. To recognize the provisions of the Indian Information Technology Act, 2000 and its amendments, including the IT Act 2008.
3. To understand cyber laws, digital evidence, electronic commerce security, and legal aspects of cyberspace.
4. To learn various cyber security standards, compliance frameworks, governance practices, and the role of Artificial Intelligence in cyber security.

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

1. Understand various types of cybercrimes, cyber threats, cyber offences, and security challenges in electronic business environments.
2. Analyze cyber threats, attack techniques, social engineering methods, and security management practices for protecting digital assets.
3. Explore cyber laws, the Information Technology Act, digital evidence, privacy regulations, and legal frameworks governing cyberspace.
4. Apply knowledge of cyber security standards, compliance frameworks, governance practices, and emerging AI-enabled security technologies to address contemporary cyber security challenges

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Fundamentals of Cyber Security, Cyber Crime and Emerging Technologies** Introduction to Cyber Security, Cyber Crime and Cyber Law**,** History and Evolution of Cyber Crime**,** Hacking and Data Theft**,** Cyber Terrorism**,** Virus, Worms and Malware, Email Bombing, Online Gambling and Pornography, Software Piracy**,** Electronic and Digital Signatures**,** Phishing and Identity Theft**,** Introduction to Artificial Intelligence and its role in Cyber Security	07
2	**Cyber Attacks, Threats and Security Mechanisms** Password Cracking**,** Keyloggers and Spyware**,** Steganography**,** Denial of Service (DoS) and Distributed DoS (DDoS) Attacks**,** SQL Injection**,** Buffer Overflow Attacks**,** Attacks on Wireless Networks**,** Web Defacement and Web Jacking**,** Attack Vectors and Attack Surface**,** AI-based Threat Detection and Intrusion Detection Systems	07
3	**Cyber Offenses, Social Engineering and Threat Intelligence** Cyber Offenses and Investigation, Social Engineering Techniques, Cyber Stalking, Cyber Café and Cyber Crimes, Botnets and their Operations, How Criminals Plan Cyber Attacks, Cyber Threat Analysis, Dynamic and Deliberate Targeting, Threat Actors and Threat Intelligence, AI-assisted Threat Hunting and Cyber Forensics	07
4	**Electronic Commerce Security and Digital Transactions** Evolution and Development of E-Commerce, Policy Frameworks for Secure Electronic Business, Paper-Based versus Paperless Contracts, E-Commerce Models (B2B, B2C, C2C), E-Security Concepts, E-Payment Mechanisms, Card-Based Payment Systems, E-Cheque, E-Cash, E-Payment Threats and Protection Mechanisms, Security for E-Commerce, AI-based Fraud Detection and Risk Assessment in Digital Payments.	07
5	**Cyber Laws, IT Act and Legal Frameworks** Cyber Crime and Criminal Justice, Information Technology Act 2000, Information Technology (Amendment) Act 2008, Penalties, Adjudication and Appeals under the IT Act, Contract Aspects of Cyber Law, Security Aspects of Cyber Law, Intellectual Property Rights in Cyberspace, Electronic Evidence and Digital Forensics, Criminal Aspects of Cyber Law, Legal and Ethical Issues of Artificial Intelligence, AI Governance and Responsible AI.	07
6	**Security Standards, Compliance and Emerging Trends** IT Security versus IT Compliance, Governance, Risk Management and Compliance (GRC), Critical Security Controls, Cyber Security Standards and Frameworks, ISO/IEC 27001, NIST Cyber Security Framework (CSF), PCI-DSS, SOX, GLBA, HIPAA, OWASP Top Ten, GDPR, CIS Controls, Security of AI Systems, Adversarial Attacks on AI Models, Generative AI in Cyber Security, Future Trends in Cyber Security and Digital Regulations.	07
	Total	42

Books Recommended:

Textbooks:

1. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India, 2023 Edition.
2. Information Systems Security, Wiley India, 2021 Edition.
3. Cybersecurity – Attack and Defense Strategies, Packt Publishing, 4th Edition, 2024.
4. Cryptography and Network Security: Principles and Practice, Pearson Education, 9th Edition, 2022.
5. Cybersecurity Essentials, Wiley, 2022 Edition.
6. Artificial Intelligence for Cybersecurity, CRC Press, 2024.

Reference Books:

1. Practical Cybersecurity Architecture, O'Reilly Media, 2023.
2. The NICE Cybersecurity Workforce Framework: Cybersecurity Intelligence and Analytics, Springer, 2022.
3. E-Commerce Security and Privacy, Springer, 2022 Reprint Edition.
4. Cyber Law and Cyber Crimes, Snow White Publications, Latest Edition.
5. The Indian Cyber Law, Bharat Law House, Latest Edition.
6. The Information Technology Act, 2000 with Amendments and Allied Rules, Universal Law Publishing / Professional Book Publishers, Latest Edition.
7. Cyber Security and Global Information Assurance, Information Science Publishing, Latest Edition.

Web Resources :

1. https://www.meity.gov.in, (Ministry of Electronics and Information Technology, Government of India)
2. https://www.cert-in.org.in (Indian Computer Emergency Response Team – CERT-In)
3. https://legislative.gov.in (Information Technology Act, 2000 and related cyber laws)
4. https://www.nist.gov/cyberframework (NIST Cybersecurity Framework and Security Guidelines)
5. https://owasp.org (OWASP Security Resources and OWASP Top 10)
6. https://www.sans.org/white-papers (SANS Institute Security Research and Compliance Resources)
7. https://www.iso.org/isoiec-27001-information-security.html (ISO/IEC 27001 Information Security Management Systems)
8. https://www.cisecurity.org/controls (Center for Internet Security (CIS) Controls and Best Practices)

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering										Semester : I	
Course: Agile Frameworks										Course Code: D26XOE514	
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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	--
3	--	--	3	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				--	--	--	--	--		--	

Pre-requisite: Knowledge of

1. Software Engineering

Objectives:

1. To focus on the phases of agile project management.
2. To equip the student on the scaling techniques for agile projects.
3. To analyze the performance of agile projects.
4. To develop the skills of the students on product development.
5. To equip the students on agile delivery and risk mitigation.

Outcomes:

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

1. Understand and Demonstrate basic knowledge in Software Engineering
2. Summarize the concepts of agile practices and business objectives and phases of agile development framework.
3. Have an exposure on the scaling factors and models to be developed for agile projects.
4. Acquire knowledge on the agile performance measurement.
5. Develop the product based on agile factors with risk mitigation.
6. Describe the role of agile in enterprise management and incremental delivery.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction to Agile Frameworks:** **1.1** Agile definitions and historical context, Agile Values and Principles found in the Agile Manifesto, Misconceptions about Agile **1.2** Selecting an Approach that Fits: Choosing between an Agile or Traditional Approach, Selecting the Right Agile Approach	06
2	**Agile Methodologies:** **2.1** The Agile Methodologies: Common Themes, Methodology Descriptions, Extreme Programming, Scrum, Feature Driven Development, The Crystal Methodologies, Adaptive, Software Development, Dynamic Systems Development Method, Lean Software Development, Starting Monday: Investigate Further	06
3	**Extreme Programming (XP):** **3.1** Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts, Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility, Practicing XP - Thinking - Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives	07
4	**Planning Agile Projects:** **4.1** Planning for Agile Teams, Scrum Teams, XP Teams, General Agile Teams, Collaboration Rooms, Team Distribution **4.2** Agile Project Lifecycles, Typical Agile Project Lifecycles, Activities within each Phase, create product vision, Producing a Minimum Marketable Feature **4.3** Release Planning, Creating the Product Backlog, User Stories, Prioritizing and Estimating, Creating the Release Plan **4.4** Monitoring and Adapting, Task Boards and Information Radiators, Control Limits, Variance and Trend Analysis, Managing Risks and Issues, Retrospectives	10
5	**Agile Estimations and Leading Agile Teams** **5.1 Introduction** to Agile Estimations, Needs, Stakeholders, Estimation Stages, Estimation Styles and Process. Velocity, Sprint Velocity **5.2 Skills** needed by Agile Leaders, Emotional Intelligence, Listening Skills, Command and Control vs. Servant Leadership, Adaptive Leadership, Collaboration, Facilitation, Problem Solving and Participatory Decision-Making Skills, Coaching and Mentoring Teams, Conflict Resolution	07
6	**Advanced Emerging Techniques and Case Studies** **6.1** Learn, value streams and Kanban models, Lean, Crystal, DevOps and continuous deployment strategies, Scaling agile processes, Case study	06
	Total	42

Books Recommended:

Text books:

1. Roger Pressman, "Software Engineering: A Practitioner's Approach", McGraw-Hill Publications 7th Edition.
2. The art of Agile Development, James Shore and Shane Warden, 11th Indian Reprint, O'Reilly, 2018

References Books:

1. Ugrasen Suman, "Software Engineering-Concepts and Practices", Cengage Learning
2. Learning Agile, Andrew Stellman and Jennifer Greene, O'Reilly, 4th Indian Reprint, 2018
3. Practices of an Agile Developer, Venkat Subramaniam and Andy Hunt, SPD, 5th Indian Reprint, 2015
4. Agile Project Management - Jim Highsmith, Pearson Low price Edition 2004

Web Resources (For our Reference):

1. https://www.xpand-it.com/blog/top-5-agile-methodologies/
2. https://apc01.safelinks.protection.outlook.com/GetUrlReputation

Prepared by Checked by Head of the Department Principal

Program: First Year M. Tech. Computer Engineering							Semester : I				
Course: Design of Experiments							Course Code: D26XOE515				
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	Assignment	Total	
				60			15	15	10	40	100
				Laboratory Examination			Term work			Total Term work	--
3	--	--	3	Oral	Practical	Oral & Practical	Laboratory Work	Tutorial / Mini project / presentation/ Journal			
				--	--	--	--	--		--	

Pre-requisite: Knowledge of

1. Applied Statistics.
2. Regression and Analysis of Variance.

Objectives:

1. To understand the issues and principles of Design of Experiments (DOE).
2. To list the guidelines for designing experiments.
3. To become familiar with methodologies that can be used in conjunction with experimental designs for robustness and optimization.

Outcomes:

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

1. Plan data collection, to turn data into information and to make decisions that lead to appropriate action.
2. Apply the methods taught to real life situations.
3. Plan, analyze, and interpret the results of experiments.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction:** Strategy of Experimentation, Typical Applications of Experimental Design, Guidelines for Designing Experiments, Response Surface Methodology.	06
2	**Fitting Regression Models:** Linear Regression Models, Estimation of the Parameters in Linear Regression Models.	06

	Hypothesis Testing in Multiple Regression, Confidence Intervals in Multiple Regression, Prediction of new response observation, Regression model diagnostics, Testing for lack of fit.	
3	**Two-Level Factorial Designs and Analysis:** The 2^2 Design, The 2^3 Design, The General 2^k Design, A Single Replicate of the 2^k Design, The Addition of Center Points to the 2^k Design, Blocking in the 2^k Factorial Design, Split Plot Designs.	**08**
4	**Two-Level Fractional Factorial Designs and Analysis:** The One-Half Fraction of the 2^k Design, The One-Quarter Fraction of the 2^k Design, The General 2^{k-p} Fractional Factorial Design, Resolution III Designs, Resolution IV and V Designs, Fractional Factorial Split-Plot Designs.	**08**
5	**Conducting Tests:** Testing Logistics, Statistical aspects of conducting tests, Characteristics of good and bad data sets, Example experiments, Attribute Vs Variable data sets.	**08**
6	**Taguchi Approach:** Crossed Array Designs and Signal-to-Noise Ratios, Analysis Methods, Robust design examples.	**06**
	Total	**42**

Books Recommended:

Reference Books:

1. Raymond H. Mayers, Douglas C. Montgomery, Christine M. Anderson-Cook, Response Surface Methodology: Process and Product Optimization using Designed Experiment, 3rd edition, John Wiley & Sons, New York, 2001
2. D. C. Montgomery, Design and Analysis of Experiments, 5th edition, John Wiley & Sons, New York, 2001
3. George E P Box, J Stuart Hunter, William G Hunter, Statics for Experimenters: Design, Innovation and Discovery, 2nd Ed. Wiley
4. W. J. Dimond, Practical Experiment Designs for Engineers and Scientists, John Wiley and Sons Inc. ISBN: 0-471-39054-2
5. Design and Analysis of Experiments (Springer text in Statistics), Springer, A. M. Dean, and D. T. Voss

Prepared by Checked by Head of the Department Principal

<table>
<tr><td colspan="8">Program: First Year M. Tech. Computer Engineering</td><td colspan="5">Semester : I</td></tr>
<tr><td colspan="8">Course: Operations Research</td><td colspan="5">Course Code: D26XOE516</td></tr>
<tr><td colspan="4" rowspan="2">Teaching Scheme
(Hours / week)</td><td colspan="9">Evaluation Scheme</td></tr>
<tr><td colspan="3">Semester End Examination Marks (A)</td><td colspan="4">Continuous Assessment Marks (B)</td><td rowspan="2">Total marks (A+ B)</td></tr>
<tr><td rowspan="3">Lectures</td><td rowspan="3">Practical</td><td rowspan="3">Tutorial</td><td rowspan="3">Total Credits</td><td colspan="3">Theory</td><td>Term Test 1</td><td>Term Test 2</td><td>Assignment</td><td>Total</td></tr>
<tr><td colspan="3">60</td><td>15</td><td>15</td><td>10</td><td>40</td><td>100</td></tr>
<tr><td colspan="3">Laboratory Examination</td><td colspan="3">Term work</td><td rowspan="2">Total Term work</td><td rowspan="3">--</td></tr>
<tr><td>3</td><td>--</td><td>--</td><td>3</td><td>Oral</td><td>Practical</td><td>Oral & Practical</td><td>Laboratory Work</td><td colspan="2">Tutorial / Mini project / presentation/ Journal</td></tr>
<tr><td></td><td></td><td></td><td></td><td>--</td><td>--</td><td>--</td><td>--</td><td colspan="2">--</td><td>--</td></tr>
</table>

Pre-requisite: Knowledge of

1. Fundamental concepts of Mathematical statistics

Objectives:

1. To formulate a real-world problem as a mathematical programming model.
2. To understand the mathematical tools that are needed to solve optimization problems.
3. To use mathematical software to solve the proposed models.

Outcomes:

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

1. Convert a real-world problem in to a Linear Programming Problem and Interpret the solution obtained using Simplex method or other algorithms.
2. Understand reasons of formation of queues, classify various queuing systems and Apply performance parameters defined for various queuing systems for decision making in real life situations.
3. Describe concept of simulation and Apply Monte Carlo Simulation technique to systems such as inventory, queuing and Develop solutions for them.
4. Explain the need for replacement of components or machines in most economical way and Infer optimal replacement age.
5. Identify the decision situations which vary with time and Analyse them using principle of dynamic programming to real life situations.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Linear Programming Problem** Introduction to Operations Research (OR), Decision situations, Decision making process, Concept of Optimization, Mathematical Models. Linear Programming: Linear Programming Problem - Mathematical Formulation, Finding Optimal solution using Graphical method, Simplex method, Big-M method, Two Phase method, Special cases, Principle of Duality.	09
2	**Special Cases of LPP** Transportation problem: Formulation - Finding Optimal solution, Degeneracy. Assignment problem: Formulation - Finding Optimal solution. Travelling Salesman Problem.	07
3	**Dynamic Programming** Introduction - Bellman's Principle of optimality - Applications of dynamic programming to capital budgeting, inventory, employment smoothening, cargo loading and shortest path problem.	08
4	**Game Theory** Introduction - Minimax (Maximin) Criterion and optimal strategy - Solution of games with saddle points – 2 x 2 games - dominance principle - m x 2 & 2 x n games, Iterative Method.	06
5	**Queuing Model** Introduction - Poisson arrivals - Exponential service time. Single Channel – Single server - Infinite population and finite population models, Multichannel - Single server - Infinite population models. Constant Service rate - Single Channel – Single server - Infinite population.	06
6	**Simulation** Definition - Methodology of simulation – Monte Carlo Simulation Technique - applications to Inventory and Queuing problems – Advantages and Limitations of Simulation. Simulation Languages.	06
	Total	42

Books Recommended:

Reference Books:

1. Taha, H.A. "Operations Research - An Introduction", Prentice Hall, (7th Edition), 2002.
2. Ravindran, A, Phillips, D. T and Solberg, J. J. "Operations Research: Principles and Practice", John Willey and Sons, 2nd Edition, 2009.
3. Hiller, F. S. and Liebermann, G. J. "Introduction to Operations Research", Tata McGraw Hill, 2002.
4. Operations Research, S. D. Sharma, KedarNath Ram Nath-Meerut.
5. Operations Research, KantiSwarup, P. K. Gupta and Man Mohan, Sultan Chand & Sons.

Prepared by Checked by Head of the Department Principal