COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

PROGRAMME STRUCTURE & SYLLABUS

[2024 ADMISSION ONWARDS]

M.TECH. COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND SOFTWARE ENGINEERING)

SYLLABUS FOR
OUTCOME-BASED EDUCATION

MASTER OF TECHNOLOGY (M.TECH.) COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE AND SOFTWARE ENGINEERING)

(2024 admission onwards)

Program Outcomes (PSO) for M.Tech. Computer Science and Engineering (Artificial Intelligence and Software Engineering)

After the completion of M.Tech. programme, the students will be able to:

PO1: Elicit deeper and current knowledge through research/exploration leading to development work with a motivation to solve practical problems.
PO2: Communicate effectively through well-written technical documentation as well as audio-visual presentations.
PO3: Recognize the importance of entrepreneurship and innovation to create value and wealth.
PO4: Acquire mastery in the topic of study at an exceedingly higher level.

Program Specific Outcomes (PSO) for M.Tech. Computer Science and Engineering (Artificial Intelligence and Software Engineering)

After the completion of M.Tech. programme, the students will be able to:

PSO1: Attain comprehensive understanding of advanced theories and models in Computer Science, Artificial Intelligence, and Software Engineering.

PSO2: Design and integrate robust, scalable, and maintainable software systems and applications using AI components, including microservices architectures and cloud-native technologies.

PSO3: Implement software development process, lead a software development team and manage software projects.

PSO4: Enhance research skills and conduct independent research, which could lead to technological innovations and improvements in the field of AI and Software Engineering.

DEPARTMENT OF COMPUTER SCIENCE
PROGRAMME STRUCTURE AND SYLLABUS (2024 ADMISSIONS)

M.TECH. COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND SOFTWARE ENGINEERING)

Semester - I

Sl. No.	Course code	Course Title	Core/ Elective	Credits	Lec.	Lab/ Tutorial	Marks
1	24-502-0101	Mathematics for Computing	C	4	4	2	100
2	24-502-0102	Artificial Intelligence and Machine Learning	C	4	4	2	100
3	24-502-0103	Design and Analysis of Algorithms	C	4	4	2	100
4	-	Elective I	E	3	4	2	100
5	-	Elective II	E	3	4	2	100
Total for Semester I				**18**	20	10	500

Electives
24-502-0104: Cloud Computing and Internet of Things
24-502-0105: Big Data Analytics
24-502-0106: Software Quality Management
24-502-0107: Quantum Computing
24-502-0108: Advanced Data Mining
24-502-0109: Information Retrieval and Web search
24-502-0110: Social Network Analytics

Semester - II

Sl. No.	Course code	Course Title	Core/ Elective	Credits	Lec.	Lab/ Tutorial	Marks
1	24-502-0201	Software Architecture and Design Thinking	C	4	4	2	100
2	24-502-0202	Agile Software Engineering	C	4	4	2	100
3	24-502-0203	Deep Learning	C	4	4	2	100
4	24-502-0204	Seminar	C	1	0	3	100
5	-	Elective III	E	3	4	2	100
6	-	Elective IV	E	3	4	2	100
Total for Semester II				**19**	20	13	600

Electives
24-502-0205: Blockchain Technology
24-502-0206: Explainable Deep learning Models
24-502-0207: Generative AI models
24-502-0208: Soft computing
24-502-0209: Complex Networks: Theory and Applications
24-502-0210: Advances in Databases

Semester - III

Sl. No.	Course code	Course Title	Core/ Elective	Credits	Lec.	Lab/ Tutorial	Marks
1	24-502-0301	Elective - MOOC	E	2	0	10	100
2	24-502-0302	Internship *	C	1	0	0	100
3	24-502-0303	Dissertation & Viva Voce	C	15	0	20	100
Total for Semester III				**18**	0	30	300

Semester - IV

Sl. No.	Course code	Course Title	Core/ Elective	Credits	Lec.	Lab/ Tutorial	Marks
1	24-502-0401	Dissertation & Viva Voce	C	**17**	0	30	100

Total credits for Degree: **72** Total Marks: **1500**

*The students should complete the **Course 24-502-0302 : Internship** during the Vacation period (May-June)

24-502-0101: Mathematics for Computing

Core/Elective: **Core** Semester: **1** Credits: **4**

Course Description

This course introduces the study of mathematical structures that are fundamentally discrete in nature. The course is intended to cover the main aspects which are useful in studying, describing and modeling of objects and problems in the context of Linear Algebra, computer algorithms and programming languages.

Course Outcomes (CO)

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

CO	**Couse Outcome Statement**	**Cognitive Level**
CO1	Analyse the different methods for proving the correctness of the theorems and problems.	Analyse
CO2	Understand and apply the basic concepts of Linear Algebra.	Apply
CO3	Understand and apply the basic aspects of Descriptive statistics.	Apply
CO4	Understand and apply the fundamentals of probability theory.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	3	-	2
CO2	3	3	-	-	2	2	-	2
CO3	3	3	-	-	2	2	-	2
CO4	3	3	-	-	3	2	-	2

Course Content

1. Introduction – proofs – propositions – predicates and quantifiers – truth tables – first order logic – satisfiability – pattern of proof – proofs by cases – proof of an implication – proof by contradiction – proving iff – sets – proving set equations – Russell's paradox – well-ordering principle – induction – invariants – strong induction – structural induction
2. Vectors-Coordinate system-vector addition-vector multiplication-Linear combinations, span, and basis vectors-Matrix multiplication as composition-Three-dimensional linear transformations-The determinant-Inverse matrices, column space and null space- Nonsquare matrices as transformations between dimensions-Dot products and duality-Cross products-Cross products in the light of linear transformations-Cramer's rule-Change of basis-Eigenvectors and eigenvalues-vector spaces
3. Descriptive statistics: histogram, sample mean and variance, order statistics, sample covariance, sample covariance matrix – Frequentist statistics: sampling, mean square error, consistency, confidence intervals,

parametric and non-parametric model estimation

4. Probability theory: probability spaces, conditional probability, independence – Random variables: discrete and continuous random variables, functions of random variables, generating random variables – Multivariate random variables: joint distributions, independence, generating multivariate random variables, rejection sampling – Expectation: Mean, variance and covariance, conditional expectation

5. Random process: definition, mean and autocovariance functions, iid sequences, Gaussian and Poisson process , random walk – Convergence of random process: types of convergence, law of large numbers, Central limit theorem, monte carlo simulation – Markov chains: recurrence, periodicity, convergence, markov-chain monte carlo- Gibbs sampling, EM algorithm, variational inference

References

1. Bronson, R., Costa, G.B., Saccoman, J.T. and Gross, D.,Linear algebra: algorithms, applications, and techniques.4e, 2023.
2. Eric Lehman, F Thomson Leighton, Albert R Meyer, Mathematics for Computer Science, 1e, MIT, 2010.
3. Susanna S. Epp, Discrete Mathematics with Applications, 4e, Brooks Cole, 2010.
4. Gary Chartrand, Ping Zhang, A First Course in Graph Theory, 1e, Dover Publications, 2012. in
5. John Tsitsiklis. 6.041SC Probabilistic Systems Analysis and Applied Probability. Fall 2013. Massachusetts Institute of Technology: MIT OpenCourseWare. https://ocw.mit.edu
6. Albert Meyer. 6.844 Computability Theory of and with Scheme. Spring 2003. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu.
7. Michael Mitzenmacher and Eli Upfal; Probability and Computing, 2ed, Cambridge University Press, 2017

Online Resources: Course notes of Carlos Fernandez-Granda, DS-GA 1002: Probability and Statistics for Data Science https://cims.nyu.edu/~cfgranda/pages/DSGA1002_fall17/index.html

24-502-0102: Artificial Intelligence and Machine Learning

Core/Elective: **Core** Semester: **1** Credits: **4**

Course Description

Machine learning is programming computers to optimize a performance criterion using example data or past experience. This course is to discuss many methods that have their bases in different fields: statistics, pattern recognition, neural networks, artificial intelligence, signal processing, control, and data mining. Major focus of the course is on the algorithms of machine learning to help students to get a handle on the ideas, and to master the relevant mathematics and statistics as well as the necessary programming and experimentation.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Understand and explain the different types of the learning process, and key ethical considerations.	Understand
CO2	Learn to effectively prepare data for machine learning models through data cleaning, feature selection, and dimensionality reduction.	Apply
CO3	Implement and interpret linear and non-linear regression models, while comparing various classification techniques including tree-based, kernel, and ensemble methods.	Apply
CO4	Gain practical knowledge in identifying data clusters using various algorithms and discovering hidden patterns through association rule learning.	Analyze
CO5	Understand the basic building blocks of neural networks, implement the backpropagation algorithm, and explore the concept of MDPs and Q-learning.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	1	1	2
CO2	3	-	-	-	3	1	1	2
CO3	3	-	-	3	3	1	3	2
CO4	3	-	-	3	3	1	2	2
CO5	3	-	-	3	3	1	3	2

Course Content

1. Introduction to AI - What is AI? A Brief History of AI - Different types of AI - Applications of AI - Problem Solving Methods – Heuristics. Knowledge Representation and Reasoning - Planning and Decision-Making: Ethics and Societal Impact of AI.
2. Machine Learning Fundamentals - Concept of Machine Learning: Definition, applications, types of learning (supervised, unsupervised, reinforcement) - Hypothesis Spaces and Inductive Bias - Learning Process- Machine Learning Ethics and Bias. Data Preprocessing and Feature Engineering: Data Representation - Data Preprocessing - Features and Types - Dimensionality Reduction – Feature Identification - Feature selection – Feature extraction - Feature Importance-High dimensional data and Manifolds.
3. Regression and Classification - Regression: Linear Regression – Non-Linear regression – evaluation metrics for regression– Classification: Binary, multi-class, and multi-label classification – lazy learners - tree-based techniques - kernel-based techniques - probabilistic techniques - and ensembled techniques – evaluation metrics for classification.
4. Clustering and Rule Mining - Clustering: Partitioning based – hierarchical based – density based– grid-based – model based - Rule mining: Apriori algorithm, FB Growth - association rules. Outlier Detection - LOF.
5. Artificial Neural Networks and Reinforcement Learning -Neural Networks: McCulloch-Pitts neurons, Hebb's networks, Hopfield networks, Boltzmann machines, Perceptrons, multilayer perceptrons, backpropagation. Reinforcement Learning: Markov Decision Processes (MDPs), Q-learning.

References

1. Ethem Alpaydin, Introduction to Machine Learning, 3e, MIT Press, 2014
2. Tom M. Mitchell, Machine Learning, McGraw Hill Education; 1e, 2017
3. Stephen Marsland, Machine Learning, An Algorithmic Perspective, 2e, CRC Press, 2015
4. Giuseppe Bonaccorso, Machine Learning Algorithms, 1e, Packt Publishing Limited, 2017
5. Ethem Alpaydin, Machine Learning- The New AI, MIT Press, 1e, 2016
6. Andrew Ng, Machine Learning Yearning, ATG AI (Draft version), 1e, 2018
7. Rohit Singh, Tommi Jaakkola, and Ali Mohammad.*6.867 Machine Learning.* Fall 2006. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu
8. Andrew Ng, https://www.coursera.org/learn/machine-learning

24-502-0103: Design and Analysis of Algorithms

Core/Elective: **Core** Semester: **1** Credits: **4**

Course Description

The course covers the foundational algorithms in depth. The course helps in understanding the working and complexity of the fundamental algorithms and to develop the ability to design algorithms to attack new problems.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Understand the basic concepts of design and analysis of fundamental algorithms.	Understand
CO2	Develop the ability to design algorithms to attack new problems.	Apply
CO3	Prove the correctness of algorithms.	Analyze
CO4	Develop the ability to analyze the complexity of algorithms.	Analyze
CO5	Understand Complexity classes, concepts of P and NP problems	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	3	3
CO2	3	-	-	3	3	-	3	3
CO3	-	3	-	3	3	-	3	3
CO4	3	3	-	-	3	-	3	3
CO5	3	-	-	3	3	-	3	3

Course Content

1. Introduction to design and analysis of algorithms, models of computation, correctness proofs, insertion sort, computational complexity, Master theorem , proof of Master theorem, merge sort, heaps, heap sort, binary search, binary search trees.

2. Graph algorithms, BFS and DFS, Dijkstra’s algorithm, proof of correctness of Dijkstra’s algorithm, Complexity analysis of Dijkstra’s algorithm , Negative weight edges and cycles , Bellman-Ford algorithm, proof of correctness and complexity of Bellman-Ford, All pairs shortest paths, Floyd- Warshall algorithm, proof of correctness and complexity, Minimum Spanning Trees , Prim’s algorithm, Cut property, Kruskal’s algorithm, proof of correctness and complexity analysis of Kruskal's Algorithm, Maximum-Flow networks, Ford-Fulkerson method, proof of correctness and complexity, Edmonds-Karp algorithm

3. Probability review, Experiments, outcomes, events, Random variables, Expectation, Linearity of Expectation,

Indicator Random Variables, Hiring Problem, Quicksort , Best case and Worst case complexity, Randomized Quicksort , Average case complexity , Hashing, Chaining, Open Addressing, Universal Hashing, Perfect Hashing , Analysis of hashing operations

4. Dynamic Programming , Rod-cutting problem, Recursive formulation, Bottom-up reformulation of recursive algorithms, Optimal Substructure Property, Matrix chain multiplication, Complexity of dynamic programming algorithms, Sequence Alignment , Longest common subsequence, Greedy algorithms, Optimal substructure and greedy-choice properties , 0-1 and fractional Knapsack problems, Huffman coding

5. P vs NP, NP Hardness, Reductions, Travelling Salesman Problem, NP-Completeness, SAT, 2- SATand 3-SAT, Vertex Cover

References

1. Michale T Goodrich and Roberto Tamassia, Algorithm Design and Applications, Wiley, 2014
2. Thomas H. Cormen et al, Introduction to Algorithms, MIT Press; 4th edition 2022.
3. Jon Kleinberg, Eva Tardos, Algorithm Design, Pearson; 1st edition August 2013.
4. Robert Sedgewick, Kevin Wayne, Algorithms, Addison Wesley; 4th edition 2011.
5. Steven S. Skiena, The Algorithm Design Manual, Springer; 3rd ed. October 2020

24-502-0104: Cloud Computing and Internet of Things

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

Cloud computing is an overarching concept where computation has moved from on premise to public/private servers commonly addressed as Cloud. Internet of Things is a specialized area of computing having an interdisciplinary nature. In this course, the advent of virtualization, later transformation from on-premise distributed computing to cloud computing and different models provided will be discussed. IoT deals with edge devices communicating with the cloud and it requires knowledge on lightweight communication protocols, data visualization, etc. We learn to create IoT solutions based on public platforms, discuss its security vulnerabilities and remedies.

Course Outcomes (CO)

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand various basic concepts related to cloud computing technologies.	Understand
CO2	Explore cloud technologies, architectures, and standards	Understand
CO3	Design methodologies and do programming for building cloud applications.	Apply
CO4	Understand the basic concepts of design of Internet of Things.	Understand
CO5	Develop the engineering skills to build internet of things solutions.	Apply
CO6	Understand security vulnerabilities of cloud and apply solutions	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	-	3	-	-	-
CO3	-	3	-	3	-	3	3	-
CO4	3	-	-	3	-	-	3	-
CO5	-	-	-	3	-	2	-	-
CO6	1	-	-	3	-	2	-	-

Course Content

1. Cloud Computing Overview: On-premise computing, client-server model, Distributed computing. Virtualization: virtual machines, hypervisor, full and para virtualization. Benefits of cloud computing, Edge and fog computing. Automated build management, deployment and monitoring of applications. Containers and docker.

2. Cloud architecture: Layers in cloud architecture, Hosting and management of applications. Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS). Scalability and reliability in cloud. Examples for each model. Web and Mobile applications communicating with cloud. Microservices vs Monolithic architectures. Applications of cloud computing healthcare, smart homes, smart grid, etc.

3. Introduction to Internet of Things. Embedded Systems: Sensors and actuators, SCADA, Cyber Physical Systems. Devices for IoT. Gateways and Routers. Time series data, measurement and acquisition of data. MQTT Protocol: broker, publish-subscribe, Mosquitto. Connectivity for IoT: Ethernet, Wireless, Bluetooth Low Energy, Zigbee, Mobile, LoRA, RFID.

4. Programming for IoT: Hosting IoT server in cloud. Using public Cloud IoT platforms: AWS/Azure/Google Cloud Etc. IoT Core, storing data, Analytics using public IoT platforms. Communicating from edge devices to gateway: Programming for Arduino/Raspberry Pi or Python compatible boards. Data Representation and Visualization: Building dashboards and mobile apps for IoT Analytics. Applications of IoT: Diagnostics, Maintenance and Predictive Analytics.

5. Cloud security: Authentication and Authorization, Tokens, API Key, Identity and Access Management in cloud. Threat analysis for IoT: Types of Cyber Attacks on cloud and IoT and techniques to prevent such attacks. Securing IoT and Cloud: Encryption of data, symmetric and asymmetric key encryption. Digital Signatures and certificates.

References

1. Toby Velte, Anthony Velte, Robert Elsenpeter: Cloud Computing, A Practical Approach, 1e, McGraw-Hill Education, 2009.
2. Rajkumar Buyya, James Broberg, Andrzej Goscinski: Cloud Computing: Principles and Paradigms, 1e, Wiley, 2013.
3. Giacomo Veneri and Antonio Capasso, Hands-On Industrial Internet of Things: Create a powerful Industrial IoT infrastructure using Industry 4.0, 1st Edition, Packt Publishing, 2018.
4. Mayur Ramgir, Internet of Things: Architecture, Implementation and Security, 1st Edition, Pearson, 2019.
5. R. Buyya, S N. Srirama, Fog and Edge Computing: Principles and Paradigms, Wiley Series on Parallel and Distributed Computing, 1st Edition, Wiley, 2019.
6. Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, 2nd Edition, MIT Press, 2017.

24-502-0105: Big Data Analytics

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

Big Data concerns large-volume, complex, growing data sets with multiple, autonomous sources. With the fast development of networking, data storage, and the data collection capacity, Big Data is now rapidly expanding in all science and engineering domains. The traditional data mining algorithms also need to be adapted for dealing with the ever-expanding datasets of tremendous volume.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Build practical skills on developing MapReduce jobs.	Apply
CO2	Explain the search algorithms that perform efficiently on massive datasets.	Understand
CO3	Explain the algorithms for data stream processing.	Understand
CO4	Explain the link analysis methods in the context of social networks and PageRank algorithms.	Understand
CO5	Demonstrate the power of some of the online algorithms for massive data.	Apply
CO6	Explain the randomized, approximate and one-pass algorithms for mining from massive datasets.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	3	-	-	3	-
CO2	3	-	-	-	3	-	3	-
CO3	3	-	-	-	3	-	3	2
CO4	3	-	-	-	-	-	3	3
CO5	3	-	-	3	-	-	3	-
CO6	3	3	-	-	3	-	3	-

Course Content

1. Introduction to MapReduce – the map and reduce tasks, MapReduce workflow, fault tolerance. - Algorithms for MapReduce – matrix multiplication, relational algebra operations- Complexity theory for MapReduce

2. Locality-Sensitive Hashing - shingling of documents, min-hashing. Distance measures, nearest neighbors,

frequent itemsets- LSH families for distance measures, Applications of LSH- Challenges when sampling from massive data

3. Mining data streams – stream model, stream data sampling, filtering streams – bloom filters, counting distinct elements in a stream - Flajolet-Martin algorithm. Moment estimates - Alon-Matias-Szegedy algorithm, counting problems for streams, decaying windows

4. MapReduce and link analysis- PageRank iteration using MapReduce, topic-sensitive-PageRank - On-line algorithms – Greedy algorithms, matching problem, the adwords problem – the balance algorithm
5. Computational model for data mining – storage, cost model, and main memory bottleneck. Hash based algorithm for mining association rule – improvements to a-priori, park-chen-yu algorithm, multistage algorithm, approximate algorithm, limited-pass algorithms – simple randomized algorithm, Savasere, Omiecinski, and Navathe algorithm, Toivonen algorithm

References

1. Jure Leskovec, Rajaraman, A., & Ullman, J. D., Mining of Massive Datasets, Cambridge University Press, 2e, 2016
2. Charu C. Aggarwal, Data Streams: Models and Algorithms, 1e, Springer, 2007
3. Michael I Jordan et.al , Frontiers in Massive Data analysis, 1e, National Academies Press, 2013
4. Nathan Marz & James Warren, Big Data: Principles and best practices of scalable realtime data systems, Manning Publications, 2015

24-502-0106: Software Quality Management

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

This course discusses basic software project quality management principles and techniques as they relate to software project planning, monitoring and control. This course describes the basics of software verification and validation planning with an emphasis on software peer reviews and software testing. The course also covers software configuration management, technical metrics for software.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the basics and benefits of software quality engineering	Understand
CO2	Plan, implement and audit a Software Quality Management program for their organization	Apply
CO3	Select, define, and apply software measurement and metrics to their software products, processes and services	Apply
CO4	Understand the fundamentals of the configuration management process to include configuration identification, configuration control, status accounting, and audits	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	-	-	-	2	2	-	-	-
CO2	1	-	-	1	1	2	1	-
CO3	2	-	-	2	2	1	-	1
CO4	-	-	-	2	2	1	-	-

Course Content

1. Introduction to software quality: Software Quality - Hierarchical models of Boehm and McCall - Quality measurement - Metrics measurement and analysis - Gilb's approach -GQM Model

2. Tools for Quality - Ishikawa's basic tools - CASE tools - Defect prevention and removal - Reliability models - Rayleigh model - Reliability growth models for quality assessment

3. Testing for reliability measurement Software Testing - Types, White and Black Box, Operational Profiles - Difficulties, Estimating Reliability, Time/Structure based software reliability - Assumptions, Testing methods, Limits, Starvation , Coverage, Filtering, Microscopic Model of Software Risk

4. Software reliability and availability - standards and evaluation of process - ISO 9000 - SEI Capability Maturity Model (CMM) - Software configuration management

5. Technical metrics for software - metrics for the analysis model - metrics for design model - metrics for source

code - metrics for testing - metrics for maintenance - technical metrics for object oriented systems - distinguishing characteristics - class oriented metrics -operation oriented metrics - testing metrics -project metrics

References

1. Allan C. Gillies, Software Quality: Theory and Management, 3e, Cengage, 2003
2. Ron S Kenett, E. R Baker, Software Process Quality- Management and Control, 1e, CRC, 1999
3. Stephen H. Kan , Metrics and Models in Software Quality Engineering, 1e, AW, 2014
4. Patric D. T.O connor , Practical Reliability Engineering, 5e, John Wesley & Sons, 2011
5. Roger S. Pressman, Software Engineering - A practitioner's approach, 8e, McGraw Hill,2014

24-502-0107: Quantum Computing

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

This course introduces concepts of Quantum Computing. This includes representation of quantum information and use of quantum algorithms. The course will cover design of quantum circuits and provide understanding of quantum noise and quantum cryptography.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the basic concepts of Quantum Computing.	Understand
CO2	Examine quantum representation of information and encryption.	Analyze
CO3	Design quantum circuits using qubit gates.	Apply
CO4	Apply quantum algorithms for computation.	Apply
CO5	Understand quantum noise and error correction for fault tolerant computation.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	2	1	-	-	2	-	-	-
CO2	1	-	-	-	1	1	1	-
CO3	2	2	-	2	2	1	-	1
CO4	1	2	-	1	1	-	-	-
CO5	2	1	-	2	2	2	1	1

Course Content

1. Introduction to Quantum Computation: Quantum bits, representation of a qubit and multiple qubits. Quantum mechanics, Probabilities and measurements, entanglement, density operators and correlation, Measurements in bases other than computational basis.
2. Quantum correlations: Bell inequalities and entanglement, Schmidt decomposition, super-dense coding, teleportation, PPT criterion. Quantum Circuits: single qubit operations, multiple qubit gates, Universal quantum gates, design of quantum circuits.
3. Quantum Algorithms: Classical computation on quantum computers. Relationship between quantum and classical complexity classes. Deutsch’s algorithm, Deutsch’s-Jozsa algorithm, quantum search.
4. Quantum Information and Cryptography: Comparison between classical and quantum information theory. Shannon entropy, noiseless coding, Bell states. Quantum teleportation. Quantum Cryptography, no cloning theorem.
5. Noise and error correction: Quantum noise and quantum operations, Flip code, The Shor code, Quantum error correction, Stabilizer codes, Hamming bound. fault-tolerant quantum computation.

References

1. Quantum Computation and Quantum Information, M. A. Nielsen & I. Chuang, Cambridge University Press, 10th Edition, 2010.
2. J. Hidary, Quantum Computing: An Applied Approach, 1st Edition, Springer Publishing, 2019.
3. David McMahon, Quantum computing explained, Wiley-Interscience, 1st Edition, John Wiley & Sons, Inc. Publication, 2008.
4. A. Peres. Quantum Theory: Concepts and Methods. 1st Edition, Springer, 1995.
5. J. Preskill, Lecture Notes on Quantum Information and Computation, California Institute of Technology, 2021.
6. Mark M. Wilde, Quantum information Theory, 1st Edition, Cambridge University Press, 2012.
7. D. A. Lidar and T. A. Brun, Quantum error correction, 1st Edition, Cambridge University Press, 2013.

24-502-0208: Advanced Data Mining

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

Data mining is the science of extracting hidden information from large datasets. This course offers a clear and comprehensive introduction to both data mining theory and Practice. All major data mining techniques will be dealt with and how to apply these techniques in real problems are explained through case studies.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the different techniques for analyzing data	Understand
CO2	Understand statistical descriptions of data and its visualization.	Understand
CO3	Evaluate different methods for mining frequent patterns, association and correlations in large data sets	Analyze
CO4	Evaluate different methods for data classification, clustering, outlier detection and prediction.	Analyze
CO5	Apply different spatial data mining techniques for the identification of spatial patterns.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	2	1	-	-	2	-	-	-
CO2	2	1	-	-	1	-	1	-
CO3	1	1	-	2	2	1	1	1
CO4	1	1	-	1	1	2	2	-
CO5	1	1	-	2	2	2	1	1

Course Content

1. Statistical descriptions of data-data visualization-measuring data similarity and dissimilarity-data pre-processing-data cleaning-data integration-data reduction-data transformation-data warehouse modeling-design-implementation-data cube technology- queries by data cube technology- multidimensional data analysis in Cube space
2. Mining frequent patterns, associations and correlations – pattern mining in multidimensional space- colossal patterns- approximate patterns- applications- Mining data streams-Mining Sequence patterns in transactional databases- mining sequence pattern in Biological Data
3. Classification and prediction- decision tree induction-Bayesian classification-rule-based classification- neural networks-support vector machines-lazy learners-genetic algorithms- model evaluation-Cluster analysis-portioning methods- hierarchical methods- density based methods-grid based-probabilistic model based clustering- clustering high dimensional data- constraint based clustering- clustering high dimensional data-graph clustering methods
4. Outlier detection- outliers and outlier analysis- outlier detection methods-statistical approaches- proximity

based approaches- clustering based approaches- classification based approaches-mining contextual and collective outliers- outlier detection in High-Dimensional data

5. Time series representation and summarization methods-mining time series data -Spatial data mining- spatial data cube construction-mining spatial association and co-location patterns-spatial clustering and classification methods-spatial trend analysis- Multimedia data mining-text mining- mining world wide web- trends in Data mining

References

1. Theophano Mitsa, Temporal Data mining, 1e, CRC Press, 2018
2. Jiawei Han &Micheline Kamber, Jian Pei, Data mining concepts and techniques, 1e, Elsevier, 2014
3. Shawkat Ali, A. B. M., and Wasimi, Saleh Ahmed. Data Mining: Methods and Techniques. Australia, Thomson, 2007.

24-502-0109: Information Retrieval and Web Search

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

A coherent treatment of classical and web based information retrieval that includes web search, text classification, text clustering, gathering, indexing and searching documents and methods of evaluating systems .

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand advanced techniques for text-based information retrieval.	Understand
CO2	Understand Boolean and vector space retrieval models	Understand
CO3	Evaluate various text classification techniques	Analyze
CO4	Understand Web search characteristics, web crawling and link analysis	Understand
CO5	Build working systems that assist users in finding useful information on the Web	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	1	-	-	-	2	-	-	-
CO2	1	-	-	-	1	-	-	-
CO3	2	-	-	2	2	1	-	1
CO4	1	-	-	-	1	-	-	-
CO5	2			2	2	2	1	1

Course Content

1. Taxonomy of IR Models – Classic models- Set theoretic model- Algebraic models- Probabilistic model- Structured text retrieval models- Models for browsing- Retrieval evaluations-Reference collections
2. Query languages-query operations-text and multimedia languages-Text operations-document preprocessing-matrix decompositions and latent semantic indexing-text compression –indexing and searching-inverted files-suffix trees- Boolean queries-sequential searching-pattern matching
3. Text Classification, and Naïve bayes-vector space classification-support vector machines and machine learning on documents-flat clustering –hierarchical clustering
4. Web search basics-web characteristics-index size and estimation- near duplicates and shingling-web crawling-distributing indexes- connectivity servers-link analysis-web as a graph- PageRank-Hubs and authorities- question answering
5. Online IR systems- online public access catalogs-digital libraries-architectural issues-document models -representations and access- protocols

References

1. Ricardo Baezce Yates, BerthierRibeiro-Neto , Modern Information Retrieval: The Concepts and Technology behind Search, 3e, ACM Press, 2017
2. Christopher D. Manning, PrabhakarRaghavan and HinrichSchütze , Introduction to Information Retrieval, 1e, Cambridge University Press, 2008
3. Bruce Croft, Donald Metzler and Trevor Strohman, Search Engines: Information Retrieval in Practice, 1e, AW, 2009

24-502-0110: Social Network Analytics

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

The course on Social Network Analytics offers an in-depth exploration of the theory, methods, and applications of analyzing social networks. Social networks have become a powerful lens through which to understand human behavior, relationships, and information flow in various contexts including online platforms, organizations, communities, and societies. This course delves into the principles and techniques used to study, model, and analyze social networks, along with their practical applications in diverse domains.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the concept of semantic web and related applications.	Understand
CO2	Learn knowledge representation using ontology.	Understand
CO3	Understand human behavior in the social web and related communities.	Analyze
CO4	Learn visualization of social networks.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	1	-	-	-	1	-	-	-
CO2	1	-	-	-	1	-	-	-
CO3	1	-	-	1	1	-	-	-
CO4	1	-	-	-	1	1	-	-

Course Content

1. Introduction to Semantic Web: Limitations of current Web - Development of Semantic Web - Emergence of the Social Web - Social Network analysis: Development of Social Network Analysis - Key concepts and measures in network analysis - Applications of Social Network Analysis.
2. Ontology and their role in the Semantic Web: Ontology-based knowledge Representation - Ontology languages for the Semantic Web: Resource Description Framework - Web Ontology Language - Modelling and aggregating social network data: State-of-the-art in network data representation - Ontological representation of social individuals - Ontological representation of social relationships - Aggregating and reasoning with social network data.
3. Extracting evolution of Web Community from a Series of Web Archive - Detecting communities in social networks - Definition of community - Evaluating communities - Methods for community detection and mining - Applications of community mining algorithms - Tools for detecting communities social network infrastructures and communities

4. Understanding and predicting human behaviour for social communities - User data management - Inference and Distribution - Enabling new human experiences - Reality mining - Context - Awareness - Privacy in online social networks - Trust in online environment - Trust models based on subjective logic - Trust network analysis - Trust transitivity analysis - Combining trust and reputation - Trust derivation based on trust comparisons - Attack spectrum and countermeasures.

5. Graph theory - Centrality - Clustering - Node-Edge Diagrams - Matrix representation - Visualizing online social networks, Visualizing social networks with matrix-based representations - Matrix and Node-Link Diagrams - Hybrid representations - Applications - Cover networks - Community welfare - Collaboration networks - Co-Citation networks.

References

1. Peter Mika, ―Social Networks and the Semantic Web‖, First Edition, Springer 2007.
2. Borko Furht, ―Handbook of Social Network Technologies and Applications‖, 1st Edition, Springer, 2010.
3. Guandong Xu ,Yanchun Zhang and Lin Li, ―Web Mining and Social Networking – Techniques and applications‖, First Edition, Springer, 2011.
4. Dion Goh and Schubert Foo, ―Social information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively‖, IGI Global Snippet, 2008.
5. Max Chevalier, Christine Julien and Chantal Soulé-Dupuy, ―Collaborative and Social Information Retrieval and Access: Techniques for Improved user Modelling‖, IGI Global Snippet, 2009.
6. John G. Breslin, Alexander Passant and Stefan Decker, ―The Social Semantic Web‖, Springer, 2009.

24-502-0201: Software Architecture and Design Thinking

Core/Elective: **Core** Semester: **2** Credits: **4**

Course Description

This course introduces the essential concepts of software architecture. Software architecture is an abstract view of a software system distinct from the details of implementation, algorithms, and data representation. Architecture is, increasingly, a crucial part of a software organization's business strategy. Here we discuss how architecture is conceived and implemented and how design thinking plays a role in development of a software product.

Course Outcomes (CO)

CO	Course Outcome Statement	Cognitive Level
CO1	Understand need for better software architecture and factors affecting it	Understand
CO2	Explore pattern oriented design and different types of software architectures	Analyze
CO3	Understand design concepts and software models in cloud and mobile domains.	Understand
CO4	Design products based on various design thinking approaches and representations	Apply
CO5	Understand case studies of products and develop architectures based on design thinking	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	2	-	-	-	2	-	1	-
CO2	1	-	-	-	2	1	1	-
CO3	-	2	-	-	2	2	1	-
CO4	-	2	-	2	1	-	1	1
CO5	-	-	2	2	1	-	1	1

Course Content

1. Software life cycle, Influences on Software architecture from stakeholders. Software Quality attributes, Guiding Quality Design Decisions. Designing for Availability, Interoperability, Modifiability, Performance, Security, Testability, and Usability. Designing Architecture: Architecturally Significant Requirement, Attribute driven design method, Views. Programming practises: Object oriented design.
2. Pattern oriented design: categories, relationship and Layers. Pipes and Filters architectural pattern, Blackboard, Distributed systems, Broker, Hardware-Software codesign. Model-View-Controller. Presentation-abstraction-control, Microkernel, Reflection. Whole-part design, Master-slave design, command processor, view handler, Forwarder-receiver, client-dispatcher-server, publisher-subscriber.
3. Architectures in Cloud and Mobile: On premise vs cloud, Mobile architectures: Android and IOS based application design. Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS). Multi-tenant implementation of software. Databases: RDBMS, NoSQL, timeseries, Redundancy. Use of Frameworks in coding. SOLID principles.
4. Design thinking: Understand, Define, Ideate, Prototype, Test. Double diamond approach, Research in Design Thinking: Quantitative vs Qualitative. Roles in design thinking: Equalizer, Archaeologist, Interpreter, Devil’s advocate. Design of Business Strategy: Circle of influence, customer experience, Services and delivery.

Metrics for Design Thinking: Enquiries, Sales, referrals, impression, etc.

5. Design Thinking representations: Empathy map, Affinity diagram, mind map, journey map. Marketing. Designing for change and growth. Disruptive innovation, Blue ocean strategy, Cold start problem, tipping point, . Case studies in design thinking.

References

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, Third Edition, Pearson, 2012.

2. Frank Buchmann, Regine Meunier et. al.: Pattern-oriented Software architecture, A system of patterns, Volume 1, Wiley, 2008.

3. Beverly R Ingle: Design Thinking for Entrepreneurs and Small business, Apress, 2013.

4. Tim Brown: Change by Design: How Design Thinking Transforms Organizations and Inspire Innovations, Harper Collins, 2019.

5. Documenting Software Architectures, Views and Beyond, 2nd Edition. P. Clements, F. Bachmann, L. Bass, et. al. Adison-Wesley SEI series, 2010.

6. Nigel Cross, Design Thinking: Understanding how Designers Think and Work, Bloomsbury Visual Art, 2019.

24-502-0202: Agile Software Engineering

Core/Elective: **Core** Semester: **2** Credits: **4**

Course Description:

Computer software has become pervasive in our commerce, culture, transportation, medical, telecommunications, military, industrial, entertainment, office and our everyday activities. Software engineering is important because it enables us to model, design and build complex software systems in a timely manner and with high quality. This course discusses the processes, methods and tools for effective software development and project management in the agile way.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Create a software product architecture using UML	Apply
CO2	Communicate with the development team using industry standard notations, designs and documentations.	Apply
CO3	Estimate the cost of a software project and apply various techniques, metrics and strategies for testing software projects.	Analyze
CO4	Work as a team leader by establishing goals and forming teams.	Apply
CO5	Understand the user requirements and plan the development work using agile project management principles.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	2	-	-	2	2	-	1	-
CO2	-	2	-	-	2	-	2	-
CO3	2	-	-	2	2	-	1	-
CO4	-	1	1	-	-	-	3	-
CO5	1	-	-	2	1	1	2	-

Course Content

1. Emergence of Software Engineering, Software design notations, Object-Oriented Analysis and Design using Unified Modelling Language (UML), Use Case Model Development, Object and Class Diagrams, Interaction Diagrams, Sequence models, Activity Diagrams, State Chart Diagrams, Package diagrams
2. Software Life Cycle Models, Waterfall Model, Prototyping Model, Spiral Model, Software Requirements Specification, SRS Document, Function-oriented Design, , Scheduling, Critical Path Method, PERT Charts, Gantt Charts, Organization and Team Structures
3. Metrics for Project Size Estimation, COCOMO Model, Software Quality, Software Quality Management System, Testing Concepts and Terminologies, Black-box Testing, White-Box Testing, Statement Coverage, Branch Coverage, Path Coverage, McCabe's Cyclomatic Complexity Metric, Software Maintenance.
4. Agile Principles, Variability and Uncertainty, Work in Process, Progress, Performance, Scrum Framework, Scrum Roles, Responsibilities & Characteristics of Product Owner, ScrumMaster, Development Team,

Sprints, Timeboxing, Sprint Planning, Sprint Execution

5. Product Backlog, Good Product Backlog Characteristics, Requirements and User Stories, Characteristics of Good Stories, Estimation and Velocity, PBI Estimation Units, Planning Poker, Scrum Planning Principles, Product Planning (Envisioning), Portfolio Planning, Release Planning, Sprint Planning

References

1. Agile: An Essential Guide to Agile Project Management, The Kanban Process and Lean Thinking + A Comprehensive Guide to Scrum, James Edge, Bravex Publications, 2020

2. Agile Software Engineering, Orit Hazzan and Yael Dubinsky, Springer London, 2008

3. Object Oriented Modeling and Design with UM, James Rumbaugh and Michael Blaha, Pearson Education India, 2nd edition, 2007

4. Software Engineering: A Practitioner's Approach, Roger Pressman, McGraw Hill, 9th Edn 2023

5. Fundamentals of Software Engineering, Rajib Mall, PHI Learning, 5th Edition 2018

6. Scrum: The Art of Doing Twice the Work in Half the Time, Jeff Sutherland and J.J. Sutherland, Currency; First Edition, 2014

7. Essential Scrum: a practical guide to the most popular agile process, Kenneth S. Rubin, Addison-Wesley, 2012

8. Effective Project Management: Traditional, Agile, Extreme, Hybrid, 8th Edition, Robert K. Wysocki; Wiley, 2019

9. Clean Code: A Handbook of Agile Software Craftsmanship, Robert C. Martin, Pearson, First edition 2008

24-502-0203: Deep Learning

Core/Elective: **Core** Semester: **2** Credits: **4**

Course Description:

Deep learning is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. This course describes deep learning techniques used by practitioners in industry, including deep feedforward networks, regularization, optimization algorithms, convolutional networks, sequence modeling, and practical methodology. This course is useful to students planning careers in either industry or research, and for software engineers who want to begin using deep learning in their products or platforms

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the need for Deep learning, Feed forward networks, Learning XOR, Gradient based Learning, Hidden units.	Understand
CO2	Differentiate between training error and generalization error, Underfitting and Overfitting. And Identify Regularization strategies, Dataset Augmentation, Adversarial Training.	Analyze
CO3	Describe the working of Convolution Operation, Sparse interactions, Parameter sharing, Equivariant representations, Pooling and Recurrent Neural Networks	Understand
CO4	Understand different types of Autoencoders, Undercomplete Autoencoders, Regularized Autoencoders, Dimensionality Reduction.	Understand
CO5	Explain Deep generative models like Boltzmann Machines, Restricted Boltzmann Machines.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	1	3	-	-	-
CO3	3	2	-	1	3	-	-	-
CO4	3	-	-	1	3	-	-	-
CO5	3	2	-	1	3	-	-	-

Course Content:

1. Deep Networks: Feed forward networks – Learning XOR- Gradient based Learning – Hidden units – Architecture design- Back propagation – Differentiation algorithms
2. Regularization for Deep Learning: Penalties-Constrained optimization-Under constrained problems- Dataset

augmentation-Semi Supervised learning- Sparse representation- Adversarial training- Optimization for training deep models: Basic algorithms-Algorithms with adaptive learning rates

3. Convolutional Networks: Convolution-Pooling-Variants of pooling- Efficient convolutional algorithms – Recurrent and Recursive Nets: Recurrent Neural Networks-Deep Recurrent Networks- Recursive Neural Networks- Explicit memory
4. Linear Factor Models: Probabilistic PCA- ICA – Slow feature analysis – Sparse coding – Autoencoders: UndercompleteAutoencoders – Regularized Autoencoders- Learning Manifolds- Applications of Autoencoders – Representation learning
5. Deep generative models: Boltzmann Machines – RBM- Deep Belief Networks-Deep Boltzmann Machines- Convolutional Boltzmann Machines- Directed generative Nets

References

1. Nithin Buduma, Nikhil Buduma and Joe Papa, Fundamentals of Deep Learning, 2nd Edition, O’Reilly, 2022
2. Jon Krohn and Grant Beyleveld, Deep learning Illustrated, Addison-Wesley; 1st edition, 2019
3. M Gopal, Deep Learning, Pearson Education; 1st edition, 2022
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 1e, MIT Press, 2016
5. Josh Patterson and Adam Gibson, Deep Learning: A Practitioner's Approach, 1e, Shroff/O'Reilly, 2017

24-502-0204: Seminar

Core/Elective: **Core** Semester: **2** Credits: **1**

Course Description

The student has to prepare and deliver a presentation on a research topic suggested by the department before the peer students and staff. They also have to prepare a comprehensive report of the seminar presented.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Identify, read, and interpret an academic research article from the literature that is related to his/her academic area of interest and present it before the committee.	Analyze
CO2	Organize and communicate technical and scientific findings effectively in written and oral forms.	Apply
CO3	Demonstrate the academic discussion skills to emphasize, argue with clarity of purpose using evidence for the claims.	Evaluate

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	2	-	1	2	-	-	-
CO2	-	2	-	-	2	-	-	-
CO3	3	2	-	1	2	-	-	-

24-502-0205: Blockchain Technology

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

This course intends to provide a comprehensive insight into various Blockchain techniques. The objectives are to give an insightful introduction to the basic concepts of blockchain and its applications in various domains.

Course Outcomes (CO)

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

CO	Course Outcome Statement	Cognitive Level
CO1	Understand the fundamentals of blockchain technology	Understand
CO2	Understand the essentials of Bitcoin and beholding bitcoins as blockchains	Understand
CO3	Analyze and design the Ethereum Blockchain	Analyze
CO4	Build Factom Blockchains	Apply
CO5	Analyze the powers of blockchains and their applications in various domains	Analyze
CO6	Study the impact of blockchains on industry	Analyze
CO7	Execute a mini project on blockchain	Apply

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	-	-	-	1	3	-	-	-
CO2	-	-	-	1	3	-	-	-
CO3	-	-	-	1	-	3	-	-
CO4	-	-	-	1	-	3	3	-
CO5	3	-	-	1	-	3	3	-
CO6	-	-	3	1	3	3	-	-
CO7	3	-	3	3	-	3	3	-

Course Content

1. Introduction to blockchain: Structure of blockchains, Blockchain life cycle, working of a blockchain, picking a blockchain, exploring blockchain applications, building trust with blockchains, Blockchain in action: Use cases, introducing bitcoin blockchains.
2. Bitcoin & Ethereum blockchains: Understanding bitcoins, comprehending bitcoins as blockchains, analyzing Ethereum blockchains, introducing ripple and factom blockchains and their importance
3. Powerful blockchain platforms: Getting introduced to Hyperledger, Hyperledger vision, Hyperledger sawtooth, understanding the blockchain fabric, understanding business, and smart blockchains, IBM Blockchains, Stellar: an optimized blockchain

4. Industry impacts of blockchains: Blockchains in financial technology, Blockchains in various industries such as insurance, Government, Real-estate, health care, Telecommunication, Transportation, etc..
5. Case Study and mini-project: Studying different blockchain projects as a case study and submit a report and present the work, Designing a blockchain application as a mini-project, and presenting the work.

References

1. Blockchain and Crypto Currency, Editors: Makoto YanoChris DaiKenichi MasudaYoshio Kishimoto, 1st Edition, Springer, 2020.
2. Blockchain or Dummies, Tiana Laurence, 1st Edition , John Wiley & Sons, Inc, , 2017.
3. Blockchain Blueprint for a new economy, Melanie Swan, 1st Edition,O'Reilly, 2017.
4. Blockchain Technology: Applications and Challenges, Panda, S.K., Jena, A.K., Swain, S.K., Satapathy,
5. S.C. , 1st Edition, Springer, 2021
6. Blockchain and Distributed Ledgers, Alexander Lipton and Adrien Treccani, 1st Edition, World Scientific Press, 2021

24-502-0206: Explainable Deep Learning Models

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description:

This course equips students with the essential skills and knowledge to navigate the complex landscape of Explainable Deep Learning Models (EDLMs). Students learn to apply a range of interpretability techniques, including feature importance analysis, rule-based models, and model-agnostic methods. By applying explainability techniques to real-world datasets and challenges, students develop practical expertise. Additionally, they gain a awareness of the ethical implications surrounding AI transparency and interpretability, preparing them to navigate these issues responsibly in their future endeavors.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Apply interpretability techniques to enhance the transparency of machine learning models.	Apply
CO2	Evaluate and compare different interpretability methods.	Analyze
CO3	Understand the trade-offs between model complexity and interpretability in diverse scenarios.	Analyze
CO4	Apply explainability techniques to real-world datasets and challenges.	Apply
CO5	Gain awareness of the ethical implications related to AI transparency and interpretability.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	-	-	3	-
CO2	3	-	-	1	-	-	-	3
CO3	3	2	-	1	3	-	-	-
CO4	3	-	-	1	-	-	3	-
CO5	3	2	-	1	3	-	-	-

Course Content:

1. Introduction to Explainable AI - Science of Interpretable Machine Learning, Motivation, Challenges, and Mythos of Model Interpretability, Human Factors in Explainability, Interpreting Interpretability, XAI flow,Making ML models Explainable: Intrinsic Explanations, Post Hoc Explanations,Global or Local Explainability, Properties of Explanations.

2. Intrinsic Explainable models: Loss Function, Linear Regression, Logistic Regression, Decision Trees, KNN. Model Agnostic Methods For XAI: Global Explanations, Local Explanations, shap.KernelExplainer, Text Explainer, Gradient Explainer, Local Linear Surrogate Models (LIME): mathematical representation,

Bagging classifier, Boosting classifier, Decision Tree, Extra Trees, Creating Lime Explainer, SHAP for Boosted Trees

3. Post hoc Explanations - Explaining the Predictions of Any Classifier, Pitfalls, Challenges, and Evaluation of Feature Attributions, OpenXAI, The Disagreement Problem in Explainable Machine Learning, Counterfactual Explanations (or) Algorithmic Recourse, Learning Model-Agnostic Counterfactual Explanations for Tabular Data

4. Attention and Concept Based Explanations - Quantifying Interpretability of Deep Visual Representations, Interpretability Beyond Feature Attribution, Data Attribution and Interactive Explanation, Equitable Valuation of Data, Explainable Active Learning (XAL), Theory of Explainability and Interpreting Generative Models

5. Explainability for Fair Machine Learning - Connections with Robustness, Privacy, Fairness, and Unlearning, Right to Explanation and the Right to be Forgotten, Fairness via Explanation Quality, Mechanistic Interpretability and Compiled Transformers, Understanding and Reasoning in Large Language Models

References

1. Explainable AI with Python, Antonio Di Cecco and Leonida Gianfagna, Springer
2. Hands-On Explainable AI (XAI) with Python: Interpret, visualize, explain, and integrate reliable AI for fair, secure, and trustworthy AI apps, Denis Rothman, Packt publisher
3. Interpretable Machine Learning, by Christoph Molnar https://christophm.github.io/interpretable-ml-book/84
4. Interpretable Machine Learning with Python: Learn to build interpretable high performance models with hands-on real-world examples, by Serg Masís , Packt publisher

24-502-0207: Generative AI models

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

This course provides an overview of generative AI, covering foundational concepts, understanding how AI models can create new content, major models, practical applications like image and text generation and even discussing about prompt engineering

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Outline the scope and application of Generative AI	Understand
CO2	Interpret different language models and their role in AI	Analyze
CO3	Analyze the concept of GPT and its variants	Analyze
CO4	Illustrating the core principles of GAN	Understand
CO5	Understanding the concept and significance of prompt engineering	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	1	3	-	-	-
CO3	3	-	-	1	3	-	-	-
CO4	3	-	-	1	3	-	2	-
CO5	3	-	-	1	3	-	-	-

Course Content

1. Introduction to Generative AI, Scope of Generative AI, Overview of generative models and their applications Importance of Generative AI in various domains, Ethical considerations and challenges
2. Introduction to language models and their role in AI, Traditional approaches to language modeling, Deep learning-based language models and their advantages, LLM architectures: RNNs, LSTMs, and Transformers
3. Understanding GPT (Generative Pre-trained Transformer) Introduction to GPT and its significance Pre-training and fine-tuning processes in GPT Architecture and working of GPT models Overview of GPT variants and their use cases
4. The core principles of GANs and their architecture. Different GAN architectures. Training and evaluating GANs, addressing common challenges. Exploring GAN applications in image generation and manipulation. Auto encoders
5. Prompt Engineering: Enhancing Model Outputs Understanding the concept and significance of prompt engineering Strategies for designing effective prompts Techniques for controlling model behavior and output quality

References

1. The Artificial Intelligence and Generative AI Bible:The Most Updated and Complete Guide" by Alger Fraley "Ripples of Generative AI: How Generative AI Impacts, Informs and Transforms Our Lives" by Jacob Emerson
2. "Neural Networks and Deep Learning" by Michael Nielsen
3. Generative Adversarial Networks" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville
4. "Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play" by David Foster
5. "Demystifying Prompt Engineering: AI Prompts at Your Fingertips (A Step-By-Step Guide)" by Harish Bhat
6. Generative AI with Python and TensorFlow 2 by Joseph Babcock and Raghav Bali
7. Generative AI with LangChain By BenAuffarth
8. Generative AI in Practice by Bernard Marr

24-502-0208: Soft Computing

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

The aim of this course is to cover fundamental concepts used in Soft computing. As part of this course the students will get exposure to Fuzzy logic, Artificial Neural Networks and optimization techniques using Genetic Algorithm. To provide hands - on practices to the students applications of Soft Computing techniques to solve a number of real life problems will be covered. This course will provide exposure to theory as well as practical systems and software used in soft computing.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Learn Fuzzy logic and its applications.	Understand
CO2	Understand the basic concepts of artificial neural networks and its applications.	Understand
CO3	Solve single-objective optimization problems using GAs.	Apply
CO4	Solve multi-objective optimization problems using Evolutionary algorithms.	Apply
CO5	Apply Soft computing techniques to solve problems in various application domains.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	1	3	-	-	-
CO3	3	2	-	1	-	-	3	-
CO4	3	2	-	1	-	-	3	-
CO5	3	-	-	1	-	-	3	-

Course Content

1. Introduction to Soft Computing: Concept of computing systems-"Soft" computing vs "Hard" computing-Characteristics of Soft computing-Some applications of Soft computing techniques.
2. Fuzzy logic: Introduction to Fuzzy logic - Fuzzy sets and membership functions - Operations on Fuzzy sets - Fuzzy relations , rules , propositions, implications and inferences - Defuzzification techniques - Fuzzy logic controller design - Some applications of Fuzzy logic.
3. Genetic Algorithms: Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques -Basic GA framework and different GA architectures - GA operators: Encoding , Crossover , Selection , Mutation , etc.-Solving single - objective optimization problems usingGas.
4. Multi-objective Optimization Problem Solving: Concept of multi-objective optimization problems (MOOPs) and issues of solving them - Multi-Objective Evolutionary Algorithm (MOEA) - Non - Pare to approaches to solve MOOPs - Pareto - based approaches to solve MOOPs Some applications with MOEAs.

5. Artificial Neural Networks: Biological neurons and its working - Simulation of biological neurons to problem solving - Different ANNs architectures - Training techniques for ANNs Applications of ANNs to solve some real life problems.

References

1. Timothy J.Ross, Fuzzy Logic with Engineering Applications, 4e, Wiley, 2016.
2. S. Rajasekaran, and G.A.Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms : Synthesis and Applications,1e, Prentice Hall India,2003.
3. Melanie Mitchell, An Introduction to Genetic Algorithms , 1e, MITPress, 1998.
4. Nikola K.Kasabov, Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering, 1e, MITPress, 1996.
5. S.N.Sivanandam and S.N.Deepa, Principles of Soft Computing, 3e, Wiley, 2018.
6. Randy L. Haupt and Sue Ellen Haupt: Practical GeneticAlgorithms, 2e, Wiley, 2004.
7. Simon Haykin: Neural Networks and Learning Machines, 3e, Pearson, 2009.
8. J.-S.R.Jang,C.-T.Sun,and E.Mizutani:Neuro Fuzzy and Soft Computing,1e,Pearson Education India, 2015.

24-502-0209: Complex Networks: Theory and Applications

Core/Elective: **Elective**   Semester: **2**   Credits: **3**

Course Description

Complex networks provide a powerful abstraction of the structure and dynamics of diverse kinds of interaction viz people or people-to-technology, as it is encountered in today's inter-linked world. This course provides the necessary theory for understanding complex networks and applications built on such backgrounds.

Course Outcomes (CO)

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

CO	Course Outcome Statement	Cognitive Level
CO1	Explain and appreciate complex networks and complex network systems as different from other network systems viz. computer networks, transportation networks, etc.	Analyze
CO2	Explain the mathematical representation of complex networks in computer programs.	Understand
CO3	Demonstrate random graph generation processes and associated properties.	Apply
CO4	Discriminate various algorithms for community detection in complex networks.	Analyze
CO5	Illustrate and explain the flow models used in complex networks for modelling social, economic, and biological systems.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	1	3	-	-	-
CO3	3	-	-	1	-	-	3	-
CO4	3	2	-	1	3	-	-	-
CO5	3	2	-	1	1	-	3	-

Course Content

1. Networks of information – Mathematics of networks – Measures and metrics – Large scale structure of networks – Matrix algorithms and graph partitioning
2. Network models – Random graphs – walks on graphs - Community discovery – Models of network formation – Small world model - Evolution in social networks – Assortative mixing- Real networks - Evolution of random network - Watts-Strogatz model – Clustering coefficient - Power Laws and Scale-Free Networks – Hubs - Barabasi-Albert model – measuring preferential attachment- Degree dynamics – nonlinear preferential attachment.
3. Processes on networks – Percolation and network resilience – Epidemics on networks – Epidemic modelling - Cascading failures – building robustness- Dynamical systems on networks – The Bianconi-Barabási model – fitness measurement – Bose-Einstein condensation

4. Models for social influence analysis – Systems for expert location – Link prediction – privacy analysis – visualization – Data and text mining in social networks - Social tagging
5. Social media - Analytics and predictive models – Information flow – Modelling and prediction of flow - Missing data - Social media datasets – patterns of information attention – linear influence model – Rich interactions

References

1. Mark J. Newman, Networks: An introduction, 1e, Oxford University Press, 2010
2. Charu C Aggarwal (ed.), Social Network Data Analytics, 1e, Springer, 2011
3. David Easley and Jon Kleinberg, Networks, Crowds, and Markets: Reasoning about a highly connected World, 1e, Cambridge University Press, 2010
4. Albert-Laszlo Barabasi, Network Science, 1e, Cambridge University Press, 2016

24-502-0210: Advances in Databases

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

This is a second course in database systems which cover advanced aspects of database systems touching upon the theoretical advancements to handle the new areas and challenges related to the management of data. The course introduces the students to the frontiers of the classical database systems and takes them to the multidimensional data and the associated processing techniques. Later, a large multitude of specialty databases are introduced. This course consolidates the theory and practices pertaining to big data storages and cloud databases.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand various aspects of physical database design and usage analysis.	Understand
CO2	Understand various aspects of Online Analytical Processing (OLAP)	Understand
CO3	Understand the concepts of Object-Based Databases.	Understand
CO4	Understand the concepts of Spatial and Temporal Data.	Understand
CO5	Construct cloud database models.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PS04
CO1	3	-	-	3	3	2	-	-
CO2	3	-	-	3	3	3	-	-
CO3	3	-	-	3	3	2	-	-
CO4	3	-	-	3	3	2	-	-
CO5	3	3	-	-	2	3	-	-

Course Content

1. Physical Database Design: The Physical Database Design Process - Data Volume and Usage Analysis - Controlling Data Integrity - Missing Data – Denormalization –Partitioning - File Organizations – Heap-Sequential-Indexed-Hashed – Non-unique indexing.

2. Online Analytical Processing: Recent Enhancements and Extensions to SQL - Analytical and OLAP

Functions–Multidimensional Analysis - New Data Types- New Temporal Features in SQL- Other Enhancements. Need for Data Warehousing – Architectures- Data Mart and Data Warehousing Environment - Real-Time Data Warehouse Architecture - Enterprise Data Model- Status/Event/Transient/Periodic Data - Derived Data - Star Schema and variations - Fact Tables - Dimension Tables - Normalization - Surrogate Key - Hierarchies - Unstructured Data.

3. Object-Based Databases: Complex Data Types - Structured Types and Inheritance in SQL - Table Inheritance - Array and Multiset Types in SQL - Object-Identity and Reference Types in SQL - Implementing O-R Features - Persistent Programming Languages - Object-Relational Mapping. Object- Oriented Databases: Motivation – Concepts and Features – Object Modelling – Indexing – Design Considerations- Object-Oriented versus Object-Relational. XML Databases: Motivation - Structure of XML Data - XML Document Schema - Querying and Transformation – XPath – XQuery – XSLT - Application Program Interfaces to XML - Storage of XML Data - XML Applications.

4. Spatial and Temporal Data: Motivation - Time in Databases - Spatial and Geographic Data - Multimedia Databases -Mobility and Personal Databases - Active Databases, Time series Databases. Advanced Transaction Processing : Transaction-Processing Monitors - Transactional Workflows - E-Commerce - Main-Memory Databases - Real-Time Transaction Systems - Long- Duration Transactions.

5. Classification of NoSQL Database Management Systems, Key-Value Stores- Document Stores- Wide-Column Stores - Graph-Oriented Databases–Redis, MongoDB, Cassandra, Neo4j – Hadoop data storage–Pig, Hive, HBase – Introduction to Integrated Data Architecture.
   Cloud Databases: Database as a service (DBaS), Amazon SimpleDB, DynamoDB – EnterpriseDB - Google Cloud SQL, Google BigQuery – Microsoft Azure SQL.

References

1. A. Hoffer Jeffrey, V. Ramesh, Topi Heikki, Modern database management, 12e, Pearson, 2015.
2. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6e,McGraw- Hill, 2013
3. SherifSakr, Big Data 2.0 Processing Systems: A Survey, 1e, Springer: Briefs in Computer Science, 2016
4. Lee Chao, Cloud Database Development and Management, 1e, CRC Press, 2013
5. Rini Chakrabarti and Shilbhadra Dasgupta, Advanced Database Management System, 1e, Dreamtech Press, 2011

24-502-0301: Elective - MOOC

Core/Elective: **Elective** Semester: **3** Credits: **2**

Course Description

A credit-based MOOC course of minimum 12 weeks duration or three non-credit based MOOC courses of 4-weeks duration from SWAYAM/NPTEL/any other platforms approved by the Department Council.

24-502-0302: Internship

Core/Elective: **Core** Semester: **3** Credits: **1**

Course Description

A minimum 1 month internship from the institute/industry approved by the Department council. Internship should be completed during the May-June summer vacation.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Develop a holistic understanding and practical skills for professional and academic success.	Apply
CO2	Demonstrate enhanced capabilities in problem-solving, effective communication, entrepreneurial thinking, and advanced subject mastery.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PS04
CO1	3	3	3	3	3	2	3	3
CO2	3	3	3	3	2	2	3	2

24-502-0303: Dissertation & Viva Voce

Core/Elective: **Core** Semester: **3** Credits: **15**

Course Description

The dissertation work spans two semesters. Through the dissertation work, the student has to exhibit the knowledge in terms of engineering or technological innovation or research ability to solve the contemporary problem. On completion of the first part of the work, the student shall submit an interim dissertation report. The qualitative and quantitative results of the work will be evaluated through a viva- voce exam.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Demonstrates in-depth knowledge and thoughtful application through the detailed analysis of the chosen research problem.	Analyze
CO2	Assesses the gap in knowledge by acquiring knowledge about previous works, their interpretation, and application.	Analyze
CO3	Demonstrates the design of the proposed methodology and its merits.	Apply
CO4	Organizes the interim dissertation content with proper structure and sequencing.	Apply
CO5	Demonstrates academic discussion skills to emphasize, argue with clarity of purpose using evidence for the claims.	Evaluate

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PS04
CO1	3	-	-	3	3	-	3	-
CO2	3	-	-	3	1	-	2	-
CO3	3	-	-	3	2	-	3	-
CO4	-	3	-	-	2	-	2	-
CO5	-	3	-	-	2	3	3	3

23-502-0401: Dissertation & Viva Voce

Core/Elective: **Core** Semester: **4** Credits: **17**

Course Description

The dissertation work spans two semesters. Through the dissertation work, the student has to exhibit the knowledge in terms of engineering or technological innovation or research ability to solve the contemporary problem. On completion of the work, the student shall submit a final dissertation report. The qualitative and quantitative results of the work will be evaluated through a viva-voce exam.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Demonstrates in-depth knowledge and thoughtful application through the detailed analysis of the problem chosen for the study	Analyze
CO2	Assesses the gap by acquiring knowledge about the previous works, and its interpretation and application	Analyze
CO3	Demonstrates the design of the proposed methodology and its merits.	Apply
CO4	Organize the interim dissertation content with proper structure and sequencing	Apply
CO5	Demonstrate the academic discussion skills to emphasize, argue with clarity of purpose using evidence for the claims.	Evaluate
CO6	Show ability to evaluate and reflect on critical questions.	Evaluate

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	3	3	3	-	-	3
CO2	3	-	3	3	3	-	-	3
CO3	3	-	-	3	3	-	-	3
CO4	-	3	-	-	3	-	-	3
CO5	-	3	-	-	3	-	-	3
CO6	3	-	-	-	3	-	-	3

Learning Outcomes and Assessment

Each course's learning outcomes will be assessed based on one or many methods, including the internal written tests, quizzes, presentations, seminars, assignments in the form of lab exercises, and group projects. The above assessment methods will be attentively created to support the intended learning outcomes that have been set out for a particular course. The program outcome attainment is measured using the CO/PO mappings.