VIDYASAGAR UNIVERSITY

Midnapore, West Bengal

PROPOSED CURRICULUM&SYLLABUS (DRAFT) OF

BACHELOR OF COMPUTER APPLICATION- BCA (HONOURS)

4-YEAR UNDERGRADUATE PROGRAMME

(w.e.f. Academic Year 2023-2024)

Based on

Curriculum & Credit Framework for Undergraduate Programmes (CCFUP), 2023& NEP, 2020

VIDYASAGAR UNIVERSITY
BACHELOR OF COMPUTER APPLICATION- BCA (HONOURS)
(under CCFUP, 2023)

Level	YR.	SEM	Course Type	Course Code	Course Title	Credit	L-T-P	Marks		
								CA	ESE	TOTAL
BCA (Hons.)	3rd	V	SEMESTER-V							
			Major-8	BCAHMJ08	T: Object Oriented Programming using JAVA; P: Practical	4	3-0-1	15	60	75
			Major-9	BCAHMJ09	T: Machine Learning; P: Practical	4	3-0-1	15	60	75
			Major-10	BCAHMJ10	T: Software Engineering; P: Practical	4	3-0-1	15	60	75
			Major Elective-01	BCAHDSE1	Artificial Neural Networks / Image Processing / Pattern Recognition	4	3-1-0/ 3-0-1	15	60	75
			Minor-5	BCAMIN05	T: Design Thinking & Innovation; P: Practical	4	3-0-1	15	60	75
			Semester-V Total			20				375
		VI	SEMESTER-VI							
			Major-11	BCAHMJ11	T: Computer Graphics; P: Practical	4	3-0-1	15	60	75
			Major-12	BCAHMJ12	T: Data Science; P: Practical	4	3-0-1	15	60	75
			Major-13	BCAHMJ13	T: Cloud Computing; P: Practical	4	3-0-1	15	60	75
			Major Elective-02	BCAHDSE2	Probability and Statistics / Social Network Analysis / Data Visualization	4	3-1-0/ 3-0-1	15	60	75
			Minor-6	BCAMIN06	T: Internet of Things(IOT) ; P: Practical	4	3-0-1	15	60	75
			Semester-VI Total			20				375
			YEAR-3			40				750
			Eligible to be awarded Bachelor of Computer Application (BCA) *on Exit*			126	Marks (Year: I+II+III)			2325

MJ = Major, MI = Minor Course, DSE = Discipline Specific Elective Course, CA= Continuous Assessment, ESE= End Semester Examination, T = Theory, P= Practical, L-T-P = Lecture-Tutorial-Practical

SEMESTER-V

MAJOR (MJ)

MJ-8: Object Oriented Programming using JAVA **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

1. To introduce the object oriented programming system concepts
2. To introduce syntax and semantics of Java programming language
3. To develop modular programs using Java
4. To setup JDK environment to create, debug and run Java programs

MJ-8T: Object Oriented Programming using JAVA **Credits 03 (45 Hrs.)**

Course Content

UNIT I:

Fundamentals of Object Oriented Programming: Basic Concepts of Object Oriented Programming (OOP), Benefits and Applications of OOP.
Java Evolution: Java Features, Difference between Java, C and C++, Java and Internet, Java Environment.
Overview of Java Language: Introduction to Simple Java Program, Use of Comments and Math function, Application of two classes, Java Program Structure, Java Tokens and statements, Implementing Java program And JVM, Command Line Arguments. **(8 hours)**

UNIT II:

Constants, Variables and Data Types: Constants, Variables, Data Types, Declaration of Variables, Giving values to Variables, Symbolic Constants, Type casting.
Operators & Expressions: Arithmetic operators, Relational operators, Logical operators, Assignment operators, Increment & Decrement operators, conditional operators, Bitwise Operators, Arithmetic Expressions, Evaluation of Expressions, Type Conversions in Expressions, Operator Precedence & Associativity.
Decision Making, Branching & Looping: Decision Making with Control Statements, Looping statements, Jump in loops, Labelled loops. **(12 hours)**

UNIT III:

Classes, Objects and Methods: Defining Class, Methods Declaration, Constructors, Methods Overloading, Overriding Methods, Inheritance
Arrays, Strings and Vectors: 1D arrays, Creating an Array, 2D arrays, Strings, Vectors,

Wrapper Classes, Enumerated Types

Inheritance: Defining, extending classes, and Implementing Interfaces. Multiple inheritance and polymorphism. **(16 hours)**

UNIT IV:

Packages: Basics of packages, System packages, Creating and accessing packages, Creating user defined packages, Adding class to a package.

Exception Handling: Using the main keywords of exception handling: try, catch, throw, throws and finally; Nested try, Multiple catch statements, Creating user defined exceptions. **(9 hours)**

Suggested Readings:

1. Balaguruswamy E. (2023). Programming with JAVA: A Primer. 7th edition. India: McGraw Hill Education
2. Schildt, H. (2022). Java: The Complete Reference. 12th edition. McGraw-Hill Education.

Reference Books

3. Arunesh Goyal, The Essentials of JAVA, Khanna Book Publishing Company Private Limited, 2012.
4. S. Malhotra and S. Choudhary, Programming in Java, 2nd Edition, Oxford University Press, 2014.

MJ-8P: Object Oriented Programming using JAVA (Lab) Credits 01 (30 Hrs.)

Course Outline:

1. Write a program to read two numbers from user and print their product.
2. Write a program to print the square of a number passed through command line arguments.
3. Write a program to send the name and surname of a student through command line arguments and print a welcome message for the student.
4. Write a java program to find the largest number out of n natural numbers.
5. Write a java program to find the Fibonacci series & Factorial of a number using recursive and non-recursive functions.
6. Write a java program to multiply two given matrices.
7. Write a Java program for sorting a given list of names in ascending order.
8. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.
9. Write a java program to read n number of values in an array and display it in reverse order.
10. Write a Java program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from

AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.

11. Create a JAVA class called Student with the following details as variables within it.
    a. USN, NAME, BRANCH, PHONE, PERCENTAGE
    b. Write a JAVA program to create n Student objects and print the USN, Name, Branch, Phone, and percentage of these objects with suitable headings.
12. Write a Java program that displays the number of characters, lines and words in a text.
13. Write a Java program to create a class called Shape with methods called get Perimeter() and getArea(). Create a subclass called Circle that overrides the getPerimeter() and getArea() methods to calculate the area and perimeter of a circle.
14. Write a Java program to create a class Employee with a method called calculateSalary(). Create two subclasses Manager and Programmer. In each subclass, override the calculateSalary() method to calculate and return the salary based on their specific roles.
15. Write a Java program for finding the cube of a number using a package for various data types and then import it in another class and display the results.
16. Write a Java program for demonstrating the divide by zero exception handling.

MJ-9: Machine Learning **Credits 04(Full Marks: 75)**

OBJECTIVE OF THE COURSE

1. Learn the basics of machine learning, understanding its uses, challenges, and various applications.
2. Build practical data skills, covering data collection, analysis, visualization, and preparation.
3. Become skilled in using classification and regression algorithms, including selecting, training, and evaluating models.
4. Dive into advanced clustering and specialized applications, using methods like K- Means, DBSCAN, and others.

MJ-9T: Machine Learning **Credits 03 (45 hrs.)**

UNIT 1: Fundamentals of Machine Learning; **[14 hours]**

Introduction to Machine Learning: What is Machine Learning? Why Use Machine Learning? , Types of Machine Learning Systems, Main Challenges of Machine Learning, Applications of Machine Learning. Why Python, scikit-learn, Essential Libraries and Tools.

UNIT 2: Data Preparation: **[12 hours]**

Working with Real Data, look at the Big Picture, Get the Data, Discover and Visualize the Data to Gain Insights, Prepare the Data for Machine Learning Algorithms, Select and Train a Model.

UNIT 3: Supervised Learning: **[12 hours]**

Classification and Regression, Some Sample Datasets, k-Nearest Neighbours, Linear Models, Naive Bayes Classifiers, Decision Trees.

UNIT 4: Unsupervised Learning: **[12 hours]**

Clustering, K-Means, Limits of K-Means, using clustering for image segmentation, Using Clustering for Preprocessing, Using Clustering for Semi-Supervised Learning, DBSCAN, Other Clustering Algorithms.

Suggested Readings:

Textbook:

1. Andreas . C. Müller and S. Guido, "Introduction to Machine Learning with Python," O'Reilly, 2017 (Unit-1)

2. Amanda . Casari and Alice . Zheng, "Feature Engineering for Machine Learning," O'Reilly Media, Inc., 2018, p. 218. (Unit-2)
3. A. Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow," O'Reilly Media, Inc., 2022.
5. Ian Goodfellow, Yoshua. Bengio, and Aaron. Courville, "Deep Learning," MIT Press, 2016. (Unit-4)

Reference Books:

1. S. Rashka and V. Mirdzhalili, "Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2," Packt, Birmingham and Mumbai, 2020.
2. S. Shalev-Shwartz and S. Ben-David, "Understanding Machine Learning: From Theory to Algorithms," Cambridge University Press, 2014.

MJ-9P: Machine Learning Lab **Credit: 1; Full Marks**: **20**

Outcome:

1. Achieve proficiency in setting up Python, installing vital libraries, and configuring essential tools.
2. Demonstrate competence in data manipulation, dataset loading, and the creation of insightful visualizations.
3. Exhibit the ability to preprocess data, address missing values, perform categorical encoding, and implement fundamental machine learning algorithms.
4. Develop an understanding of clustering techniques, create cluster visualizations, and interpret decision tree splits.

List of Programs

1. Install and set up Python and essential libraries like NumPy and pandas.
2. Introduce scikit-learn as a machine learning library.
3. Install and set up scikit-learn and other necessary tools.
4. Write a program to Load and explore the dataset of .CVS and excel files using pandas.
5. Write a program to Visualize the dataset to gain insights using Matplotlib or Seaborn by plotting scatter plots, bar charts.
6. Write a program to Handle missing data, encode categorical variables, and perform feature scaling.
7. Write a program to implement a k-Nearest Neighbours (k-NN) classifier using scikit- learn and Train the classifier on the dataset and evaluate its performance.
8. Write a program to implement a linear regression model for regression tasks and Train the model on a dataset with continuous target variables.

9. Write a program to implement a decision tree classifier using scikit-learn and visualize the decision tree and understand its splits.
10. Write a program to Implement K-Means clustering and Visualize clusters.

Datasets Link:

1. Classification Problem: https://archive.ics.uci.edu/dataset/53/iris
2. Regression Problem: https://archive.ics.uci.edu/dataset/186/wine+quality
3. Clustering Problem: https://archive.ics.uci.edu/dataset/352/online+retail

Major 10: Software Engineering **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

Students will be able to:

1. Learn software engineering process models and metrics to plan and manage software development effectively**.**
2. Learn coding best practices, inspections, and walkthroughs to produce maintainable and reliable code.
3. Learn and execute effective test strategies using white-box, black-box, and specialized testing techniques.
4. Manage software maintenance processes, including cost estimation and reverse engineering, for long-term sustainability.

MJ- 10T: Software Engineering **Credits 04 (Full Marks: 40)**

Course Content:

Unit 1: Introduction **[8L]**

Software, Software Myths, Software types, Software engineering: A Layered Technology, Different Software Process Models, Fourth Generation Techniques, Software project management.

Unit 2: Software Design **[12L]**

Problem Partitioning, Top-Down And Bottom-Up design, Decision tree, decision table, Cohesion, Coupling, Design approaches : Functional and Object- Oriented approach. Function- Oriented Software Design: SA/SD methodology, Structure analysis, DFD, Structure chart.

Unit 3: Coding **[6L]**

Coding standards, code walk- throughs, code inspection, clean room testing and documentation.

Unit 4: Software Quality Assurance **[10L]**

Quality Concepts, The Quality Movement, Software Quality Assurance, Software Reviews, Formal Technical Reviews, Formal Approaches to SQA, Statistical Software Quality Assurance, Software Reliability, Mistake Proofing for Software, Introduction to ISO standard.

Unit 5: Software Testing Technique **[8L]**

Software testing fundamentals, Test case design, White-box Testing, Basis path testing, Control structure testing, Black-box testing, Testing for specialized environments, architectures and applications.

Unit 6: Software Maintenance **[6L]**

Characteristics, reverse engineering, maintenance process models, estimation of cost.

References Books:

1. Pressman, R. S., & Maxim, B. R. (2020). *Software Engineering: A Practitioner's Approach* (9th ed.). McGraw-Hill.
2. Sommerville, I. (2016). *Software Engineering* (10th ed.). Addison-Wesley.
3. Bruegge, B., & Dutoit, A. H. (2010). *Object-Oriented Software Engineering: Using UML, Patterns, and Java* (3rd ed.). Pearson Prentice Hall.
4. Jalote, P. (2006). *An Integrated Approach to Software Engineering* (3rd ed.). Springer.
5. Mall, R. (2018). *Fundamentals of Software Engineering* (5th ed.). PHI Learning.
6. Aggarwal, K. K., & Singh, Y. (2016). *Software Engineering* (3rd rev. ed.). New Age International Publishers.
7. Braude, E. J. (2001). *Software Engineering: An Object-Oriented Perspective*. Wiley.
8. Bell, D. (2005). *Software Engineering for Students* (4th ed.). Addison-Wesley.
9. Schmuller, J. (2004). *SAMS Teach Yourself UML in 24 Hours* (3rd ed.). Sams Publishing.

MJ- 10P: Software Engineering Lab **Credit- 01** Full Marks: **20**

Sample Projects:

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
7. Software Personnel Management System
8. Credit Card Processing
9. E-book management System.
10. Recruitment system

MAJOR ELECTIVE (DSE)

Major Elective (MJ DSE) -1

(Artificial Neural Networks / Image Processing / Pattern Recognition)
Credits 04 (Full Marks: 75)

MJ DSE-1T: Artificial Neural Networks **Credit: 04 (60 Hrs.)**

Course Objectives:

1. Biological motivation to design intelligent systems and control
2. Study the learning strategies of Artificial Neural networks and their training algorithms.
3. Discuss neural networks, architectures, algorithms and applications, including Back-propagation, adaptive networks, Hopfield network
4. To acquire knowledge about associate memory and training algorithms of various associate memory networks.
5. Introducing the concepts of classical and fuzzy set theory in crisp and fuzzy logic principle, fuzziness involved in various systems
6. Study the fuzzy rule base system, decision making system and different methods of defuzzification and applications of fuzzy logic

Course Content:

UNIT I: Artificial Neural Networks: Introduction, Biological Neuron, Artificial Neuron, Basic concepts of Neural Networks, Basic Models of ANN Connections, McCulloch-Pitts Model, Characteristics of ANN, Applications of ANN. **(8 hours)**

UNIT II: Essential of ANN: Artificial Neuron Model, Operations of Artificial Neuron, Types of Neuron Activation Function, ANN Architectures, Classification Taxonomy of ANN – Connectivity, Learning Strategies (Supervised, Unsupervised, Reinforcement), Learning Rules, Numerical problems, Types of Application **(12 hours)**

UNIT III: Supervised Learning Network: Perceptron Network, Perceptron Learning Rule, Architecture, Perceptron Training Algorithm, ADALINE, MADALINE, Back Propagation Network, BP Learning Rule, Input Layer Computation, Hidden Layer Computation, Output Layer Computation **(12 hours)**

UNIT IV: Associative Memory Network: Training Algorithms for Pattern Association, Auto Associative Memory Network, Hopfield Networks. **(8 hours)**

UNIT V: Classical & Fuzzy Sets: Introduction to classical sets - properties, Operations and relations; Fuzzy sets, Membership, Uncertainty, Operations, properties, fuzzy relations, cardinalities, membership functions. Fuzzification, Membership value assignment, development of rule base and decision making system, Defuzzification to crisp sets, Defuzzification methods.

(20 hours)

TEXT BOOKS:

1. Neural Networks and Fuzzy Logic System by Bart Kosko, PHI Publications.
2. Neural Networks, Fuzzy logic, Genetic algorithms: synthesis and applications by Rajasekharan and Pai – PHI Publications.
3. Fundamental of Artificial Neural Network and Fuzzy Logic-by Rajesh Kumar, Lakshmi publications

REFERENCE BOOKS:

1. Neural Networks – James A Freeman and Davis Skapura, Pearson Education.
2. Neural Networks – Simon Hakins , Pearson Education

OR

MJ DSE-1: Image Processing **Credit: 04**

Course objectives:

1. To be familiar with Image model, sensing and acquisition, digital image representation, properties of human visual system, various applications.
2. Different image processing operations for improving image quality through enhancement, restoration and filtering etc.
3. Affine transformation and registration compressing data to save storage and channel capacity during transmission. Image segmentation for partitioning into objects and background.

MJ DSE-1T: Image Processing **Credit: 03 (45 Hrs.)**

Course Content:

UNIT-I: Introduction: Image definition and its representation, Pixels, Co-ordinate conventions, Image formats (Study of the image matrix), neighbourhood metrics, Sampling and quantization, Types of distance measure (concept only). **(8 hours)**

UNIT-II: Spatial Domain: Image enhancement techniques in spatial domain, Contrast stretching, Histogram Processing, Noise smoothing, Sharpening, Pixel Classification. **(8 hours)**

UNIT-III: Thresholding: Grey level thresholding, global/ local thresholding, Iterative thresholding, Edge detection operators, Region growing, Split/ merge techniques, Image feature/ primitive extraction, Background correction, Color enhancement. **(10 hours)**

UNIT-IV: Image restoration: Basic Framework, Interactive Restoration, Image deformation and geometric transformations, image morphing, Restoration techniques, Noise characterization, Noise restoration filters, Restoration from projections, Hough transform, Huffman coding, Segmentation **(12 hours)**

UNIT-V: Image Segmentation: Boundary detection based techniques, Point, line detection, Edge detection, Local processing, Regional processing, Region-based segmentation. **(8 hours)**

TEXT BOOKS:

1. R C Gonzalez, R E Woods, Digital Image Processing, 3rd Edition, Pearson Education.2008.

REFERENCE BOOKS:

1. A K Jain, Fundamentals of Digital image Processing, Prentice Hall of India.1989.
2. K R Castleman, Digital Image Processing, Pearson Education.1996
3. Schalkoff, Digital Image Processing and Computer Vision, John Wiley and Sons.1989.

MJ DSE-1P: Image Processing **Credit: 01 (30 Hrs.)**

All programs are implemented in Python or any other suitable software.

1. Write a program (in Python) to read and display an image.
2. Write a program (in Python) to convert a color image to gray image.
3. Write a program (in Python) to read, show, and print the properties of an image.
4. Write a program (in Python) to enlarge an image.
5. Write a program (in Python) to rotate an image in clock wise and anticlockwise direction.
6. Write a program (in Python) to implement Color conversion for RGB model to gray scale model.
7. Write a program (in Python) to perform Histogram Equalization.
8. Write a program (in Python) to calculate Histograms.
9. Write a program (in Python) to implement Sharpening (Laplacian Spatial Filters).
10. Write a program (in Python) for Smoothing and Sharpening on a color image.
11. Write a program (in Python) to implement Basic gray-level transforms.
12. Write a program (in Python) to implement Arithmetic and Logical operations on an image.

OR

MJ DSE-1: Pattern Recognition **Credit: 04 (60 Hrs.)**

Course objectives:

The objective of this course is to provide a comprehensive understanding of the principles, methodologies, and algorithms used in Pattern Recognition. Students will learn how to model, analyze, and classify patterns in data using statistical, structural, and machine learning approaches. The course focuses on both theoretical foundations and practical implementations to build intelligent recognition systems.

By the end of this course, students will be able to:

1. Understand the architecture and components of a pattern recognition system.
2. Apply statistical decision theory and learning-based methods for classification.
3. Extract and select discriminative features from raw data.
4. Implement clustering and dimensionality reduction techniques.
5. Design and evaluate recognition systems for real-world problems.

MJ DSE – 1T: Pattern Recognition **Credits: 03 (*45 Lecture*)**

Course Contents

1. Introduction to Pattern Recognition ***(07 Lectures)***

Definition and scope of pattern recognition; Examples and applications in various domains; Components of a pattern recognition system; Supervised and unsupervised learning; Concept learning and generalization; Statistical, syntactic, and neural network approaches.

2. Statistical Decision Theory ***(07 Lectures)***

Bayesian decision theory; Minimum error rate classifier; Minimum risk classifier; Class-conditional probabilities and discriminant functions; Parametric estimation – Maximum likelihood and Bayesian estimation; Non-parametric methods – Parzen windows and K-Nearest Neighbor (KNN) method.

3. Linear Classifiers and Discriminant Functions ***(08 Lectures)***

Linear discriminant functions for two classes; Perceptron learning algorithm; Gradient descent methods; Multi-class extension; Fisher's Linear Discriminant; Support Vector Machines (SVM) – basic concept and geometric interpretation.

4. Feature Extraction and Dimensionality Reduction ***(08 Lectures)***

Feature extraction techniques – statistical, structural, and signal-based; Feature selection vs. extraction; Principal Component Analysis (PCA); Linear Discriminant Analysis (LDA); Independent Component Analysis (ICA); Feature normalization and transformation.

5. Clustering and Unsupervised Learning ***(08 Lectures)***

Clustering concepts and objectives; K-Means and Hierarchical clustering; Fuzzy C-Means clustering; Self-Organizing Maps (SOM); Expectation-Maximization (EM) algorithm; Cluster validity indices; Applications of clustering in pattern recognition.

7. Recent Trends and Applications of Pattern Recognition ***(07 Lectures)***

Introduction to deep learning-based recognition; Convolutional features for classification; Hidden Markov Models (HMMs) and sequential data recognition; Pattern recognition applications in speech, biometrics, and text; Evaluation metrics – precision, recall, confusion matrix, F-measure.

MJE – 1P: Pattern Recognition Lab **Credits: 01 (30 Hrs.)**

List of Practical Exercises:

1. Implementation of Bayesian classifier for given datasets.
2. Design and implementation of K-Nearest Neighbor (KNN) classifier.
3. Application of Perceptron learning algorithm for binary classification.
4. Feature extraction from multidimensional datasets.
5. Dimensionality reduction using Principal Component Analysis (PCA).
6. Linear Discriminant Analysis (LDA) for supervised dimensionality reduction.
7. Implementation of clustering algorithms: K-Means and Fuzzy C-Means.
8. Development of a simple Support Vector Machine (SVM) classifier.
9. Evaluation of classifier performance using confusion matrix and accuracy measures.
10. Mini project: Design and testing of a pattern recognition system using real-world data (e.g., speech, handwriting, or sensor data).

Suggested Readings:

1. *Pattern Classification* – Richard O. Duda, Peter E. Hart, and David G. Stork, Wiley.
2. *Pattern Recognition and Machine Learning* – Christopher M. Bishop, Springer.
3. *Introduction to Statistical Pattern Recognition* – Keinosuke Fukunaga, Academic Press.
4. *Machine Learning* – Tom M. Mitchell, McGraw-Hill.
5. *Pattern Recognition*: Statistical, Structural and Neural Approaches – *Robert J. Schalkoff, Wiley.*

MINOR (MI)

MI – 5: Design Thinking & Innovation **Credits: 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

Operating in today's complex, dynamic, and uncertain business environment, innovation has become the primary driver of success across industries. *Design Thinking & Innovation* aims to develop learners' ability to think creatively, empathize with users, and apply systematic design-based approaches to solve real-world problems.

This course introduces students to the core principles, stages, and tools of design thinking. It helps them observe and define unstructured problems, generate creative ideas, develop prototypes, and implement innovative solutions. By the end of the course, students will be able to identify innovation opportunities, apply design methodologies to enhance customer experience, and execute human-centered innovation projects effectively.

MI – 5T: Design Thinking & Innovation **Credits: 03**

Course contents:

1. **Fundamentals of Design Thinking:** **(07 Lectures)**

   Concept and significance of innovation in business and management, Creative thinking processes and approaches to problem-solving, Design thinking mindset and objectives, Design thinking and customer-centricity, Real-world examples (Airbnb, Apple, IDEO, Netflix), Stages of design thinking – *Empathize, Define, Ideate, Prototype, Implement*.

2. **Innovation and Creativity in Context:** **(07 Lectures)**

   Difference between creativity, invention, and innovation, Role of innovation in organizational growth, Sources and types of innovation – incremental vs. radical, Barriers to creativity and innovation, Methods to overcome barriers, Leadership and culture in innovation.

3. **Empathy and Problem Definition:** **(08 Lectures)**

   Importance of empathy in innovation, Tools to build empathy (observation, interviews, journey mapping), Understanding user needs, motivations, and challenges, Observing and assimilating unstructured information, Wicked problems – definition and identification, Framing solvable design challenges using "How Might We" statements.

4. **Ideation Techniques and Tools:** **(08 Lectures)**

Ideation frameworks and creativity tools – brainstorming, mind mapping, systems thinking, Divergent and convergent thinking, Mapping customer experiences for ideation, Evaluating and selecting ideas, Group exercises and consensus building on wicked problems.

5. Prototyping and Implementation: (08 Lectures)

Concept and purpose of prototyping, Low-fidelity and high-fidelity prototypes, Rapid prototyping and iterative design, Testing and refining prototypes based on user feedback, Implementing innovative solutions, Measuring innovation impact.

6. Feedback, Redesign and Innovation Presentation: (07 Lectures)

Feedback loops and user experience evaluation, Ergonomic and user-focused design, Redesigning and recreating solutions based on feedback, Final concept testing and presentation, Showcasing innovative product or service concepts with creative design solutions.

Suggested Readings:

1. *Developing Thinking Skills (The Way to Success)* – E. Balaguruswamy, Khanna Book Publishing Company (2023)
2. *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation* – Tim Brown, Harvard Business Review Press (2008)
3. *8 Steps to Innovation* – R. T. Krishnan & V. Dabholkar, Collins Publishing

Reference Book:

1. ***Design Thinking* – Nigel Cross, Bloomsbury Publishing**

MI – 5P: Design Thinking & Innovation Lab Credits: 01

List of practical exercises:

1. Empathy Mapping: Conduct user observation and interviews to identify key insights.
2. Problem Definition: Frame unstructured issues into clear design challenges using "How Might We" statements.
3. Brainstorming Workshop: Apply creative ideation methods to generate innovative ideas.
4. Customer Journey Mapping: Visualize and analyze the customer experience and pain points.
5. Prototype Development: Create low-fidelity prototypes (sketches, mock-ups, or storyboards).
6. Rapid Prototyping and Testing: Test prototypes with users and collect feedback.
7. Feedback Analysis: Refine prototypes based on collected feedback and user insights.

8. Redesign Challenge: Re-create and improve upon an existing design or solution.
9. Innovation Pitch: Prepare and present a final innovation proposal based on the design thinking process.
10. Reflection Report: Submit a documented report highlighting all stages of the design thinking process.

SEMESTER-VI

MAJOR (MJ)

MJ-11: Computer Graphics **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

- Understand the principles and techniques used in generating and manipulating visual content
- Cover fundamental concepts such as rasterization, vector graphics, and 3D modeling.
- Learn about graphic transformations, rendering techniques, and the mathematics behind image generation.
- Emphasize the use of algorithms for image processing, animation, and visualization.
- Gain hands-on experience with graphic design tools and programming libraries like OpenGL and DirectX.
- Explore advanced topics such as ray tracing, shading models, and GPU programming.
- Develop skills to create and manipulate digital images and animations.
- Prepare for careers in game development, animation, and multimedia design.

MJ-11T: Computer Graphics (Theory) **Credits 03**

Course contents:

Module I: Basics of Computer Graphics **6 Hrs.**

Introduction, What is computer Graphics?, Area of Computer Graphics, Design and Drawing, Animation Multimedia applications, Simulation, How are pictures actually stored and displayed, Difficulties for displaying pictures.

Module II: Graphic Devices **6 Hrs.**

Cathode Ray Tube, Quality of Phosphors, CRTs for Color Display, Beam Penetration CRT, The Shadow - Mask CRT, Direct View Storage Tube, Tablets, The light Pen, Three Dimensional Devices.

Module III: C Graphics Basics **3 Hrs.**

Graphics programming, initializing the graphics, C Graphical functions, simple Programs

Module IV: Simple Line Drawing Methods **8 Hrs.**

Point Plotting Techniques, Qualities of good line drawing algorithms, The Digital Differential Analyzer (DDA), Bresenham's Algorithm, Generation of Circles

Module V: Two Dimensional Transformations and Clipping and Windowing 8 Hrs.
What is transformation?, Matrix representation of points, Basic transformation, Need for Clipping and Windowing, Line Clipping Algorithms, The midpoint subdivision Method, Other Clipping Methods, Sutherland - Hodgeman Algorithm, Viewing Transformations

Module VI: Three Dimensional Graphics 6 Hrs.
Need for 3-Dimensional Imaging, Techniques for 3-Dimesional displaying, Parallel Projections, Perspective projection

Module VII: Solid Area Scan Conversion and Three Dimensional Transformations 8 Hrs.
Solid Area Scan Conversion, Scan Conversion of Polygons, Algorithm Singularity, Three Dimensional transformation, Translations, Scaling, Rotation, Viewing Transformation, The Perspective, Algorithms, Three Dimensional Clipping, Perspective view of Cube.

Suggested Readings:

1. J.D.Foley, A.Van Dan, Feiner, Hughes Computer Graphics Principles & Practice 2nd edition
a. Publication Addison Wesley 1990.
2. D.Hearn, Baker: Computer Graphics, Prentice Hall of India 2008.
3. D.F.Rogers Procedural Elements for Computer Graphics, McGraw Hill 1997.
4. D.F.Rogers, Adams Mathematical Elements for Computer Graphics, McGraw Hill 2nd
a. edition1989.

MJ-11P: Computer Graphics Lab (Practical) Credits 01

1. Write a program to implement Bresenham‘s line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw Hermite/Bezier curve.
8. Create and rotate a triangle about the origin and a fixed point.
9. Draw a color cube and allow the user to move the camera suitably to experiment with perspective viewing.
10. To draw a simple shaded scene consisting of a tea pot on a table.
11. Develop a menu driven program to fill the polygon using scan line algorithm

12. Implementation of polygon filling using Flood-fill, Boundaryfill and Scan-line algorithms.
13. Implementation of 2D transformation: Translation, Scaling,
14. Rotation, Mirror Reflection and Shearing (write a menu driven program).
15. Implementation of Line Clipping using Cohen-Sutherland algorithm and Bisection Method.
16. Implementation of 3D geometric transformations: Translation, Scaling and rotation.
17. Implementation of Curve generation using Interpolation methods.
18. Implementation of Curve generation using B-spline and Bezier curves.
19. Write a program to draw a Hut or other geometrical figures.
20. Write a program to draw a line using DDA algorithm.
21. Write a program to draw a line using Mid-Point algorithm.
22. Write a program to draw an Ellipse using Mid-Point algorithm.
23. Write a program to rotate a Circle around any arbitrary point or around the boundary of another circle.
24. Write a menu driven program to rotate, scale and translate a line point, square, triangle about the origin.
25. .Write a program to perform line clipping.
26. .Write a program to implement reflection of a point, line.
27. .Write a program to perform shearing on a line.
28. .Write a program to implement polygon filling.

MJ-12: Data Science **Credits 3+1 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

- Understand the fundamental concepts, workflow, and applications of Data Science.
- Learn data acquisition, preprocessing, cleaning, and transformation techniques.
- Explore descriptive and inferential statistical methods for data analysis.
- Apply machine learning algorithms to solve real- world analytical problems.
- Gain practical knowledge of data visualization and storytelling skills.
- Use programming tools such as Python, NumPy, Pandas, Scikit-Learn, and Matplotlib/Seaborn.
- Develop the ability to analyze structured and unstructured data.
- Prepare students for careers in analytics, business intelligence, and research.

MJ-12 T: Data Science (Theory) **Credits 03 (45 Hrs.)**

Course contents:

Module I: Introduction to Data Science **5 Hrs.**

Definition and scope of Data Science, Data Science workflow, Applications in business, healthcare, finance, research, Types of Data, Structured vs. Unstructured Data, Role of Data Scientist.

Module II: Data Collection and Preprocessing **5 Hrs.**

Data sources, Data acquisition methods, Data cleaning (handling missing values, outliers, duplicates), Data transformation, Normalization & Standardization, Feature extraction, Feature engineering.

Module III: Descriptive and Inferential Statistics **7 Hrs.**

Measures of central tendency and dispersion, Probability distributions, Hypothesis testing, Correlation & Regression analysis, p-value, Confidence intervals, ANOVA (basic idea).

Module IV: Data Visualization **6 Hrs.**

Importance of visualization, Types of charts and plots, Data storytelling, Visualization using Matplotlib, Seaborn, and Tableau (overview).

Module V: Machine Learning Fundamentals **10 Hrs.**

Supervised vs Unsupervised learning, Train–test split, Model evaluation metrics,

- Linear Regression
- Logistic Regression
- K-Nearest Neighbors

- Decision Trees and Random Forests
- K-Means Clustering
- Basics of Deep Learning (overview)

Module VI: Data Science and Big Data Ecosystem 6 Hrs.

Big Data introduction, Hadoop ecosystem overview, MapReduce concept, Spark basics, Cloud platforms for data analytics.

Module VII: Applications and Case Studies 6 Hrs.

Business intelligence dashboards, Recommendation systems, Fraud detection, Text analytics overview, Ethical issues in data handling and privacy.

Suggested Readings:

1. Wes McKinney: Python for Data Analysis, O'Reilly Media.
2. Aurélien Géron: Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly Media.
3. Trevor Hastie, Robert Tibshirani, Jerome Friedman: The Elements of Statistical Learning, Springer.
4. Chan, Y.: Introduction to Data Mining, Pearson.
5. Montgomery & Runger: Applied Statistics and Probability for Engineers, Wiley.

MJ-12 P: Data Science Lab (Practical) Credits 01 (30Hrs.)

1. Introduction to Python for Data Science: Variables, Data Types, Functions.
2. Working with NumPy: Arrays, Mathematical operations.
3. Data manipulation using Pandas: Importing datasets, DataFrame operations.
4. Handling missing data and outliers.
5. Feature scaling and normalization techniques.
6. Data visualization using Matplotlib and Seaborn.
7. Implementing Linear Regression on real-world datasets.
8. Implementing Logistic Regression and evaluating accuracy.
9. Decision Tree and Random Forest implementation and comparison.
10. K-Means clustering on a sample dataset.
11. Explore dataset using EDA (Exploratory Data Analysis) techniques.
12. Create a simple dashboard or report for a dataset.
13. Text preprocessing and word frequency visualization (basic NLP).

Mini project: Apply the complete data science workflow on a chosen dataset.

MJ-13: Cloud Computing **Credits 3+1 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of this course is to introduce students to the fundamental concepts, architecture, and applications of Cloud Computing. It aims to develop an understanding of how cloud-based services enable scalable, on-demand, and cost-effective computing solutions for modern enterprises. Learners will explore the key service models (IaaS, PaaS, SaaS), deployment types, virtualization techniques, and storage mechanisms that form the foundation of cloud environments. The course also emphasizes emerging technologies such as serverless computing, microservices, and DevOps, along with best practices in cloud security and governance. By the end of the course, students will be able to design, deploy, and manage cloud-based solutions effectively while understanding practical implementations through case studies and hands-on exercises.

MJ-13 T: Cloud Computing **Credits: 03** ***(45 Lectures)***

Course Contents:

1. **Overview of Cloud Computing** (07 Lectures)

   Definition and essential characteristics of cloud computing; Evolution and history of cloud; Key considerations for cloud adoption; Advantages and challenges; Overview of major cloud service providers and their services (AWS, Azure, Google Cloud).

2. **Cloud Service and Deployment Models** (08 Lectures)

   Overview of cloud service models: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS); Deployment models – Public, Private, Hybrid, and Community Clouds; Cloud ecosystem and architecture overview.

3. **Virtualization and Cloud Infrastructure** (08 Lectures)

   Concept of virtualization; Types of virtualization; Hypervisors and Virtual Machines; Containers and orchestration (Docker, Kubernetes); Bare metal servers; Resource pooling and scalability; Secure cloud networking.

4. **Cloud Storage and Content Delivery Networks (CDN)** (07 Lectures)

   Basics of cloud storage; Types of cloud storage – File, Block, and Object storage; Object storage APIs; Redundancy and backup; Content Delivery Networks; Data caching and replication; Cloud database services.

5. **Emerging Trends in Cloud Computing** (08 Lectures)

Hybrid and Multi-cloud systems; Microservices architecture; Serverless computing; Cloud-native development; DevOps on the cloud; Continuous Integration and Continuous Deployment (CI/CD); Application modernization and migration.

6. **Cloud Security and Case Studies** (07 Lectures)

   Fundamentals of cloud security; Identity and Access Management (IAM); Cloud encryption and key management; Cloud monitoring and auditing; Incident response; Compliance and governance; Case studies in Google Cloud and AWS.

Suggested Readings:

1. Barrie Sosinsky, *Cloud Computing Bible*, Wiley-India, 2010.
2. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski (Eds.), *Cloud Computing: Principles and Paradigms*, Wiley, 2011.
3. Nikos Antonopoulos & Lee Gillam (Eds.), *Cloud Computing: Principles, Systems and Applications*, Springer, 2012.
4. Ronald L. Krutz & Russell Dean Vines, *Cloud Security: A Comprehensive Guide to Secure Cloud Computing*, Wiley-India, 2010.
5. Gautam Shroff, *Enterprise Cloud Computing: Technology, Architecture, Applications*, eBooks.com, 2010.
6. Toby Velte, Anthony Velte, Robert Elsenpeter, *Cloud Computing: A Practical Approach*, McGraw-Hill, 2010.
7. Dimitris N. Chorafas, *Cloud Computing Strategies*, CRC Press, 2010.

MJ-13 P: Cloud Computing Lab **Credits: 01 (30 Hrs.)**

List of Practical Exercises:

1. Study of cloud computing models (IaaS, PaaS, SaaS) through a cloud platform.
2. Create and configure virtual machines using AWS/Azure/GCP.
3. Deploy and manage a sample web application using PaaS.
4. Demonstrate file and object storage operations in cloud storage.
5. Configure a Content Delivery Network (CDN) for static content delivery.
6. Implement a simple serverless function using AWS Lambda or Google Cloud Functions.
7. Create and manage Docker containers and test container orchestration using Kubernetes.
8. Build a CI/CD pipeline for a simple cloud-hosted application.
9. Implement basic Identity and Access Management (IAM) policies.
10. Cloud monitoring and log analysis using tools such as CloudWatch or Stackdriver.
11. Case Study: Host a complete mini-project demonstrating compute, storage, and security integration on cloud.

MAJOR ELECTIVE (DSE)

Major Elective (MJ DSE) -1

(Probability and Statistics / Social Network Analysis / Data Visualization)
Credits 04 (Full Marks: 75)

MJ DSE – 2T Probability and Statistics **Credit: 04 (60 Hrs.)**

Course Objectives:

- This course aims to make the students trained to handle randomness scientifically using theory of probability.
- This course intends to make the students able to represent the statistical data in a systematic way and analyze it to draw meaningful information from them.
- Through plentiful examples and exercises, this course provides the students scope to apply probabilistic and statistical techniques to deal with the real-life problems.

Course Content:

UNIT I: **12 hours**

Basic concepts of Statistics, qualitative and quantitative data, classification of data, construction of frequency distribution, diagrammatic representation of data.
Measures of Central Tendency: Arithmetic mean, median and mode—their properties Measures of Dispersion: Range, mean deviation, quartile deviation, variance and standard deviation.

UNIT II: **08 hours**

Correlation: Definition, scatter diagram, types of correlation, measures—Karl Pearson's correlation coefficient.
Regression: Linear regression-fitting by least square method and interpretation.

UNIT III: **24 hours**

Concepts of probability: Experiment and sample space, events and operations with events, probability of an event, basic probability rules, applications of probability rules, conditional probability, Bayes theorem.
Random Variables: Discrete and continuous random variable, probability distribution of a random variable, probability mass function, probability density function, expectation and variance of a random variable.
Standard Probability Distributions: Binomial probability distribution, Poisson probability distribution, Normal probability distribution.

UNIT IV: **16 hours**

Sampling Distribution: Concept of Population and Sample, parameter and statistic, sampling distribution of sample mean and sample proportion.

Statistical Inference: Estimation and Hypothesis Testing (only concept).

Text Books

1. Manish Sharma, Amit Gupta, The Practice of Business Statistics, Khanna Book Publishing Company, 2010 (AICTE Recommended Textbook)
2. Das N. G., Statistical Methods, Combined Edition, Tata McGraw Hill, 2010.
3. Ross Sheldon M., Introduction to Probability and Statistics for Engineers and Scientists, 6th Edition, Elsevier, 2021.
4. Miller Irwin and Miller Marylees, Mathematical Statistics with Applications, Seventh Edition, Pearson Education, 2005

Reference Books

1. Pal Nabendu and Sarkar Sahadeb, Statistics: Concepts and Applications, Second Edition, PHI, 2013
2. Montgomery Douglas and Runger George C., Applied Statistics and Probability for Engineers, Wiley, 2016.
3. Reena Garg, Engineering Mathematics-I, Khanna Publishing House, 2024.
4. Reena Garg, Advanced Engineering Mathematics, Khanna Publishing House, 2023.

Web Resources

1. https://nptel.ac.in/courses/111106112
2. https://nptel.ac.in/courses/111105041

OR

MJ DSE – 2T Social Network Analysis **Credit: 04 (60 Hrs.)**

Course objectives:

- To introduce the fundamentals of Social Network Analysis and its significance in understanding societal connections and behaviors.
- To analyze various models of network growth and understand the properties of real-world networks.
- To explore link analysis algorithms and their applications in understanding relationships within a network.
- To study community detection methods and their relevance in identifying meaningful clusters within networks.

Course Content:

UNIT I: Networks and Society: What is Social Network Analysis, why do We Study Social Networks, Applications of Social Network Analysis, Preliminaries, Three Levels of Social Network Analysis, Network Basics, Node Centrality, Assortativity, Transitivity and Reciprocity, Similarity, Degeneracy. (12 hours)

UNIT II: Collection of Social Network Data: Techniques to study different aspects of OSNs — Follower-followee dynamics, link farming, spam detection, hashtag popularity and prediction, Topic Models, Modelling social interactions on the Web – Agent Based Simulations, Random Walks and variants (15 hours)

UNIT III: Security issues of Social Network: 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. (15 hours)

UNIT IV: Applied Social Data Analytics: Application of Topic models, Information Diffusion, Opinions and Sentiments – Mining, Analysis and Summarization, Case Study: Sentiment Analysis on a set of Movie Reviews using Deep Learning techniques, Recommendation Systems, Language dynamics and influence in online communities, Community identification, link prediction and topical search in social networks. (18 hours)

TEXT BOOKS:

1. Tanmoy Chakraborty, "Social Network Analysis", Wiley India Pvt. Ltd., 2021.
2. Stanley Wasserman, Katherine Faust, "Social Network Analysis: Methods and Applications", Cambridge University Press, 1994.

REFERENCE BOOKS:

1. Stanley Wasserman, Katherine Faust, “Social Network Analysis: Methods and Applications”, Cambridge University Press, 1994.

OR

MJ DSE – 2: Data Visualization **Credit: 04 (Full Marks: 75)**

Course Objectives:

1. Identify and recognize visual perception and representation of data.
2. Illustrate about projections of different views of objects.
3. Apply various Interaction and visualization techniques.
4. Analyze various groups for visualization.
5. Evaluate visualizations.

MJ DSE – 2T: Data Visualization **Credit: 03 (Full Marks: 40)**

Course Content:

UNIT-I **(10 Lectures)**

INTRODUCTION TO DATA VISUALIZATIONS AND PERCEPTION: Introduction of visual perception, visual representation of data, Gestalt principles, Information overload.

UNIT-II **(08 Lectures)**

VISUAL REPRESENTATIONS: Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications.

UNIT-III **(10 Lectures)**

CLASSIFICATION OF VISUALIZATION SYSTEMS: Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents.

UNIT-IV **(10 Lectures)**

VISUALIZATION OF GROUPS: Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization. Various visualization techniques, data structures used in data visualization

UNIT-V **(12 Lectures)**

VISUALIZATION OF VOLUMETRIC DATA AND EVALUATION OF VISUALIZATIONS: Visualization of volumetric data, vector fields, processes, and simulations, Visualization of maps, geographic information, GIS systems, collaborative visualizations, Evaluating

TEXT BOOKS:

Ward, Grinstein, Keim, Interactive Data Visualization: Foundations, Techniques, and Applications. Natick, 2nd edition,A K Peters, Ltd 2015**.**

REFERENCE BOOKS:

Tamara Munzner, Visualization Analysis & Design, 1st edition,AK Peters Visualization Series 2014
2. Scott Murray, Interactive Data Visualization for the Web 2nd Edition, 2017

MJ DSE – 2P: Data Visualization **Credit: 01 (Full Marks: 20)**

Course Outline

Module 1: Introduction to Data Visualization Tools

1. Introduction to data visualization concepts and best practices.
2. Installation and basic usage of visualization tools such as:
    - **Python libraries:** Matplotlib, Seaborn, Plotly
    - **Tableau / Power BI** (optional)

Module 2: Basic Visualization using Python

3. Create basic charts: bar charts, line plots, scatter plots, histograms, pie charts using **Matplotlib**.
4. Explore customization: titles, legends, labels, colors, and styles.

Module 3: Advanced Visualization Techniques

5. Visualizing distributions and relationships using **Seaborn**: Pair plots, box plots, violin plots, heatmaps.
6. Multivariate and categorical visualizations (e.g., hue, size, and style mapping).

Module 4: Interactive and Dashboard Visualizations

7. Interactive plots using **Plotly** or **Bokeh**.
8. Creating simple dashboards using **Plotly Dash** or **Streamlit**.

Module 5: Advanced Visualizations

9. Time-series visualization and trend analysis.
10. Geospatial data visualization using **Plotly**, **Folium**, or **GeoPandas**.
11. Network visualization basics (optional).

MINOR (MI)

MI-06: Internet of Things (IoT) **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of this course is to introduce students to the fundamental concepts, architecture, and practical applications of the Internet of Things (IoT). It aims to develop an understanding of how interconnected devices, embedded systems, and cloud-based technologies enable smart environments and data-driven decision-making. Students will learn about IoT architectures, communication models, protocols, and hardware platforms such as Arduino and Raspberry Pi. The course also focuses on IoT application development using Python and cloud integration. By the end of the course, learners will be able to design and implement basic IoT systems, apply suitable communication protocols, and understand the role of analytics and Industry 4.0 concepts in modern IoT ecosystems.

MI-6T: Internet of Things (IoT) - Theory **Credits: 03 (45 Lectures)**

Course Contents:

1. Introduction to IoT **(07 Lectures)**

Definition and characteristics of IoT; Physical and logical design of IoT; Functional blocks of IoT; IoT communication models and APIs; Applications and challenges in IoT systems.

2. IoT Architecture and Protocols **(08 Lectures)**

Machine-to-Machine (M2M) communication; Web of Things; IoT protocol stack and layered architecture; Overview of communication protocols – MQTT, CoAP, 6LoWPAN, Bluetooth Low Energy (BLE), and SDN concepts.

3. IoT Hardware Platforms **(07 Lectures)**

Overview of popular IoT hardware: Raspberry Pi, Arduino, Intel Galileo, ARM Cortex processors; Sensors and actuators; Device interfacing; Introduction to cloud computing and Fog computing for IoT.

4. Developing IoT Applications **(08 Lectures)**

Introduction to Python programming for IoT; IoT development tools and frameworks; Data acquisition and control using Python; Cloud connectivity and IoT analytics integration.

5. Case Studies and Advanced IoT Applications **(08 Lectures)**

IoT applications in smart homes, industrial automation, smart cities, healthcare, and environmental monitoring; Big Data and data visualization in IoT; Industry 4.0 and digital transformation concepts.

6. Security and Future Trends in IoT **(07 Lectures)**

IoT security issues, privacy concerns, and mitigation techniques; IoT data management and ethics; Emerging technologies – Edge computing, AI in IoT, and future research directions.

Suggested Readings:

1. Arshdeep Bahga and Vijay Madisetti, *Internet of Things: A Hands-on Approach*, Universities Press, 2015.
2. K. G. Srinivasa, G. M. Siddesh, R. R. Hanumantha, *Internet of Things*, Cengage Learning India, 2018.
3. Matt Richardson & Shawn Wallace, *Getting Started with Raspberry Pi*, O'Reilly (SPD), 2016.
4. David Hanes et al., *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things*, Pearson Education, 2017.
5. Raj Kamal, *Internet of Things: Architecture and Design Principles*, McGraw Hill Education, 2017.

MI-6P: Internet of Things (IoT) – Lab (Practical) **Credit: 01 (30 Hrs.)**

List of Practical Exercises:

1. Study of IoT components, architecture, and communication models.
2. Setting up Arduino and Raspberry Pi for IoT experiments.
3. Interfacing sensors (temperature, light, motion) and actuators with Arduino/Raspberry Pi.
4. Writing and executing basic Python scripts for IoT data acquisition.
5. Connecting IoT devices to a cloud platform (ThingSpeak / AWS IoT / Azure IoT Hub).
6. Visualizing sensor data through cloud dashboards.
7. Implementing MQTT protocol for device-to-device communication.
8. Designing a simple smart home automation prototype.
9. Developing an IoT-based environmental monitoring system.
10. Mini-project: Build and demonstrate a complete IoT application integrating hardware, software, and cloud.