



# School of Computer Science & IT Devi Ahilya Vishwavidyalaya

## SYLLABUS

### M. Tech (Computer Science) 2 years

#### Program Educational Objectives (PEOs)

- PEO 1:** Acquire theoretical and practical knowledge in Computer Science to pursue career in industry or academia.
- PEO 2:** Develop analysis, design, and implementation skills to solve real-life problems using modern tools and technologies.
- PEO 3:** Carry out research and development work leading to entrepreneurial activities.

#### Program Specific Outcomes (PSOs)

- PSO 1:** Design and implement solutions in emerging areas such as AI&ML and Data Science.
- PSO 2:** Develop expertise in cloud computing and enhance ability to undertake cloud-based projects.

## I - SEMESTER

## CS-6220: Internet Programming Using Java

### Aim:

To make students learn fundamental concept of coding and perform them practically and to develop problem-solving skills

---

### Course Outcomes:

- CO 1: Understanding of OOPs concepts clearly
  - CO 2: Understanding of client server architecture and thier working
  - CO 3: Understanding database creating and server technologies
  - CO 4: providing knowledge of advance java concepts like servlet, JSP, etc.
  - CO 5: Understanding of MVC Architecture
- 

### Unit-I

Review of java concepts: Features of Java, Object-oriented programming overview, Introduction of Java Technologies, How to write simple Java programs, Data Types, Variables, Memory concepts, control statements, looping, Method CallStack and Activation Record, Argument Promotion and Casting, Scope of declaration and Method Overloading, String Handling: The String constructors, String operators, Character Exaction, String comparison, String Buffer. Arrays: Declaring and Creating Arrays, Enhanced for Statement, Passing Arrays to Method, Multidimensional Arrays, Variable-Length Argument lists, Using Command-line Arguments

### Unit-II

Inheritance: Extending classes & related things, Packages and Interfaces: Defining a Package, Understanding CLASSPATH, Access Protection, Importing packages, creating own packages Exception Handling: Introduction, over view of doing it and keywords used, when to use it, Multithreading: What are threads, The java Thread model, Thread priorities, Thread life cycle, Thread Synchronization, Applets: Applet basics, Applet Architecture, Applet life cycle methods, Database connectivity: JDBC, The design of JDBC, Typical uses of JDBC

### Unit-III

Introduction to HTTP, webServer and application Servers, Installation of Application servers, Config files, Web.xml. JavaServlet, Servlet Development Process, Deployment Descriptors, Generic Servlet, Lifecycle of Servlet. Servlet Packages, Classes, Interfaces, and Methods, Handling Forms with Servlet, Various methods of Session Handling, various elements of deployment descriptors.

### Unit-IV

JSP Basics: JSP lifecycle, Directives, scripting elements, standard actions, implicit objects. Connection of JSP and Servlet with different database viz. Oracle, MS-SQL Server, MySQL.java.sql Package. Querying database, adding records, deleting records, modifying records, types of Statement. Separating Business Logic and Presentation Logic, Building and using JavaBean. Session handling in JSP, Types of errors and exceptions handling.

### Unit-V

MVC Architecture Introduction to Remote Method Invocation, Introduction to Enterprise Java Bean, Types of EJB, Creating and working with Session Bean

---

**Text Books:**

1. Java2:The Complete Reference by HerbertSchildt,TataMcGraw-Hill,8th Edition, 2011.
2. K.Mukhar,“BeginningJavaEE5:From Novice to Professional”,WroxPress.

**Reference Book(s):**

1. The Java Programming Language, Ken Arnold,James Gosling, David Holmes,3rd Edition, Person Education, 2000.
2. Head First Java,Kathy Sierra,Bert Bates,O’Reilly Publication,2ndEdition,2005

SCSIT, DAVV, Indore, MP - 452017

CS6418: Advanced Database Management System

**Aim:**

The aim of the course is to make students able to handle large database system (corporate database) and to be able to manipulate it efficiently.

**Course Outcomes (COs)**

**CO1:** Build strong foundation of query languages through relational algebra, calculus & QBE

**CO2:** Design conceptual, logical & physical database model through ER model and normalization.

**CO3:** Develop SQL proficiency on simple & advanced features with advanced functions like Concurrency, transaction on processing, and recovery in multi user environment

**CO4:** Acquire necessary skills for No SQL based database application development

**CO5:** Exposure to graph, spatial, and temporal databases

**Unit-I:**

**No. of Hours: 8**

Introduction: Advantages of DBMS approach, Various views of data, data independence, schema & sub-schema, primary concept of data models, database languages, transaction management, database administrator & user, data dictionary, database architectures. ER model: Basic concept, design issues, mapping constraint, keys, ER diagram, weak & strong entity-sets, specialization & generalization, aggregation, inheritance, design of ER schema, Reduction of ER Schema to tables. Domains, relation, kind of relation, Relational databases, Various types of keys: candidate, primary, alternate & foreign keys

**Unit-II:**

**No. of Hours: 8**

Relational Algebra and SQL: The structure, relational algebra with extended operations, modification of database, Idea of relational calculus, basic structure of SQL, Set operation, Aggregate functions, Null values, Nested Sub queries, derived relations, views, Modification of database, join relation, Domain, relation & keys, DDL in SQL.

**Unit-III :**

**No. of Hours: 8**

Functional dependencies & Normalization: basic definitions, Trivial & non trivial dependencies, closure set of dependencies & of attributes, Irreducible set of dependencies, introduction to normalization, non-loss decomposition, FD diagram, First, second and third normal forms, dependency preservation, BCNF, multivalve dependencies and fourth normal form, join dependencies and fifth normal form. Database Integrity: general idea, integrity rules, Domain rules, Attributes rules, assertion, triggers, integrity & SQL.

**Unit-IV:**

**No. of Hours: 8**

Transaction, Concurrency & Recovery: basic concept, ACID properties, transaction state, Implementation of atomicity & durability, Concurrent execution, Basic idea of serializability, Basic idea of concurrency control, basic idea of deadlock, Failure Classification, storage structure-types, stable storage implementation, data access, recovery & Atomicity: log based recovery, deferred database modification, immediate, Query-processing and optimization

**Unit-V**

**No. of Hours: 8**

Overview of Graph & Spatial Databases & NOSQL

**Text Books:**

1. DBMS by Sudarshan & Korth

**Reference Book(s):**

1. Internet material & eBooks for advanced topics Online Resources:

SCSIT, DAVV, Indore, MP - 452001, INDIA

## CS-4407: Information System Design

---

### Course Outcomes (COs)

**CO1:** Explain the concepts, importance and elements of designing an information system.

**CO2:** Comparative study of various process models used in information system design.

**CO3:** Demonstrate project planning and management activities in order to manage and estimate software projects.

**CO4:** Discuss structured and object-oriented analysis & design methodologies.

**CO5:** Practice object-oriented modeling using UML for the design of real life problems.

### Unit-I:

**No. of Hours: 8**

Introduction to Information System, Challenges in ISD, Elements of ISD, Roles and responsibilities in IS Design, Case Study of Information System Design; Software processes, Information system design models, A comparative study of Information System design models (Traditional, CBD, Agile and RUP process).

### Unit-II:

**No. of Hours: 8**

Project Management and Planning: Project management essentials, Project success and failures, Project Life Cycle, Project team structure and organization. Project planning: Metrics and Measurements, Project Estimation: Effort Estimation Techniques, Staffing and Personnel Planning. Project Scheduling. Software Configuration Management, and Risk Management.

### Unit-III:

**No. of Hours: 8**

Requirements engineering: Requirements Elicitation, Requirements analysis: Structured Vs. Object-oriented analysis, Prototyping analysis; Requirements specification, Requirements validation; Requirements management and its tools. Software Design: Design fundamentals, Design process, Modular design, and Component level design. Structured Vs. Object-oriented design, Refactoring.

### Unit-IV :

**No. of Hours: 8**

Object-oriented analysis and design, UML modeling, Use case diagrams: Use Case scenario and Use Case relationships, Class and Object diagrams, Sequence diagrams, Collaboration diagrams, State-Chart diagrams, Activity diagrams, Component diagrams, Deployment diagrams, Generalizations Domain Model refinements, Architecture, Packaging model elements. Case Study of large scale systems

### Unit-V:

**No. of Hours: 8**

Implementation: Traditional vs. TDD, Testing of information systems, Testing strategies, Levels of testing. Debugging, Automation Testing. Software testing tools.

### Text Books:

1. Introduction to Information System, O'Brain, Printice Hall, 1997.
2. Software Engineering: Concepts & Practices- Ugrasen Suman, Cengage Learning Publications, Second Ed. 2022.
3. Object Oriented Analysis and Design Using UML- Ugrasen Suman et al, Cengage Learning, First Edition, 2018.
4. Object Oriented Modeling and Design Using UML-James Rambaugh, Pearson Education.

### Reference Book(s):

1. Software Engineering-A practitioner's approach- R. S. Pressman, Tata McGraw-Hill International Editions, New York.

## **School of Computer Science & Information Technology, DAVV, Indore**

2. Object Oriented Analysis and Design with Applications-Grady Booch, Addison Wesley.
3. Object-Oriented Software Engineering: A Use Case Driven Approach- Ivar Jacobson et al, Pearson Education.

SCSIT, DAVV, Indore, MP - 452001, INDIA

## CS-5028: HIGH PERFORMANCE COMPUTING

---

**Aim:** To introduce the issues related to the use of HPC techniques in solving large scientific problems.

**Course Outcomes (COs):**

**CO1:** Understand the role of HPC in science and engineering.

**CO2:** Understand the architecture of GPUs and the CUDA programming model.

**CO3:** Write and optimize parallel programs using C++ and CUDA.

**CO4:** Apply GPU computing to solve high-performance computing problems.

**CO5:** Analyze and optimize GPU program performance.

**Unit-I**

**No. of Hours: 8**

Introduction to HPC: Definition and significance of HPC, History and evolution of supercomputing, Examples of HPC applications across various domains like science, engineering, finance, and healthcare, Parallel Architectures: Understanding Flynn's Taxonomy (SIMD, MIMD), Shared Memory Architectures: Concepts like multi-core processors and cache coherence, Distributed Memory Architectures: Working with clusters and interconnects (e.g., InfiniBand, Ethernet), Memory Hierarchy: Importance of caches, buffers, and optimizing data access patterns, Accelerated Architectures: Introduction to GPUs, FPGAs, and other accelerators.

**Unit-II**

**No. of Hours: 6**

Parallel programming paradigms: Shared Memory Programming (OpenMP), Concepts: threads, tasks, shared/private data, synchronization (barriers, critical sections, atomics, locks), Directives and clauses for parallelizing code, Distributed Memory Programming (MPI - Message Passing Interface), Concepts: processes, ranks, point-to-point communication, collective communication (broadcast, scatter, gather, reduce), MPI functions and their usage, Understanding different communication modes and their impact on performance, GPU Programming (CUDA/OpenCL), GPU architecture basics and programming models, Thread hierarchy, memory models, and optimization techniques (e.g., memory coalescing, shared memory usage), Using CUDA or OpenCL for developing parallel applications. Hybrid Programming: Combining MPI with OpenMP or CUDA for optimal performance on hybrid systems.

**Unit-III**

**No. of Hours: 6**

Performance analysis and optimization: Performance Metrics: Understanding metrics like FLOPs, speedup, efficiency, and scalability, Benchmarking and Profiling: Tools and techniques for evaluating and analyzing performance bottlenecks (e.g., profilers like nvprof, nvvp, memcheck), Code Optimization Techniques: Strategies for improving performance (e.g., vectorization, loop reordering, data decomposition, cache optimization), Troubleshooting and Debugging: Identifying and resolving issues in parallel programs.

**Unit-IV**

**No. of Hours: 8**

HPC systems and administration: Job Schedulers: Using tools like Slurm or LSF for managing and submitting parallel jobs on HPC clusters, Cluster Management: Basic concepts of

## School of Computer Science & Information Technology, DAVV, Indore

managing and configuring HPC clusters, High-Speed Networks: Understanding the role of network interconnects and optimizing network communication, Scalable Storage Systems: Working with parallel file systems (e.g., Lustre, GPFS) and understanding storage configurations, Cloud Computing for HPC: Leveraging cloud platforms for HPC workloads.

### Unit-V

**No. of Hours: 12**

Advanced topics (optional): Numerical Methods and Libraries: Using optimized libraries like BLAS, LAPACK, PETSc for scientific computing, Big Data Processing with HPC: Integrating HPC with big data frameworks like Hadoop or Spark, Machine Learning and Deep Learning with HPC: Accelerating AI workloads using HPC systems, Quantum Computing: Introduction to the concepts and potential applications of quantum computing.

### Text Books:

1. Kai Whang Computer Architecture and Parallel Processing McGraw-Hill Book Company
2. CUDA by Example by Jason Sanders and Edward Kandrot

### Reference Book(s):

1. John L. Hennessy and David A. Patterson. Computer Architecture: A Quantitative Approach (5th ed.). Elsevier India Pvt. Ltd. 2011.

### Online Resources:

1. Online CUDA tutorials

## CS-5715: Artificial Intelligence and Machine Learning

---

### Aim:

The learning objectives of this course are to acquaint students with the theory and principles of Artificial Intelligence and Machine Learning and with representative practical systems.

### Course Outcomes (COs)

**CO1:** Understand the basic concepts of artificial intelligence and machine learning.

**CO2:** Design and implement models based on supervised and unsupervised learning.

**CO3:** Evaluate and compare modelling aspects of various machine learning approaches.

**CO4:** Apply the ensemble methods for improving the learning algorithm.

**CO5:** Apply the concepts of neural networks and deep learning methods.

### Unit-I:

**No. of Hours: 8**

**Introduction:** Past, Present, and Future, Definition, Applications, Types of the learning system, Concept learning, Concept learning as search through a hypothesis space, General-to-specific ordering of hypotheses, finding maximally specific hypotheses, Version spaces, Inductive bias.

**Evaluation of Learning Algorithms:** Measuring the accuracy of learned hypotheses, cross-validation, learning curves, Overfitting, Bias-Variance Trade-off, Data Pre-processing, Feature Selection.

### Unit-II:

**No. of Hours: 8**

**Supervised Learning:** Linear Regression, Gradient Descent (GD), Multiple Regression, Logistic Regression, k-Nearest Neighbor (k-NN) classifier

**Decision Tree Learning:** Picking the best splitting attribute: entropy and information gain, ID3, C4.5, CART Algorithm, Decision tree classifier, Stopping Criterion & Pruning loss functions.

### Unit-III:

**No. of Hours: 8**

**Dimensionality Problem:** Dimensionality Problem, Feature Extraction, Principal Component Analysis (PCA)

**Probabilistic Classifier:** Probability and Bayes learning, Naïve Bayes classifier

**Non-probabilistic Classifiers:** Support Vector Machine (SVM) classifier.

### Unit-IV:

**No. of Hours: 8**

**Unsupervised Learning:** Clustering, k-means clustering, Hierarchical clustering

**Ensemble Methods:** Ensemble Strategies, Boosting and Bagging, Random Forest, Ada-Boost algorithm.

### Unit-V:

**No. of Hours: 8**

**Artificial Neural Network:** Introduction, Early Models, Multilayer Perceptron, Gradient Descent, Multiclass Representation, Backpropagation algorithm.

**Deep Learning:** Introduction, Convolutional Neural Networks (CNN), Recurrent Neural Networks

**Recent Trends in Deep Learning:** Generative Adversarial Networks (GAN), Large Language Model (LLM), GenAI

### Text Books:

1. Tom M. Mitchell, "Machine Learning", Tata McGraw-Hill Education, 2017.
2. Ethem Alpaydin, "Introduction to Machine Learning", PHI, Third Edition, 2015.
3. Russell and Norvig, "Artificial Intelligence: A Modern Approach" Third Edition, 201

## School of Computer Science & Information Technology, DAVV, Indore

### Reference Book(s):

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", Oreily, Third Edition, 2022
2. Andreas Muller and Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", Oreily, 2016.

### Online Resources:

1. NPTEL Course Video Lectures on "Introduction to Machine Learning" - By Prof. Sudeshna Sarkar, IIT Kharagpur
2. Coursera Course Video Lectures on "Machine Learning" - By Prof. Andrew Ng, Stanford University

SCSIT, DAVV, Indore, MP - 452001, INDIA

**Aim:**

The learning objectives of this course are to acquaint students with the theory and principles of Artificial Intelligence and Machine Learning and with representative practical systems.

**Course Outcomes (COs)**

**CO1:** Implement models based on supervised and unsupervised learning.

**CO3:** Evaluate and compare modelling aspects of various machine learning approaches using different evaluation measures.

**CO4:** Implement the ensemble methods for improving the learning algorithm.

**CO5:** Implement the concepts of neural networks and deep learning methods.

**No. of Hours: 24**

1. Write a Python program to implement a Linear Regression model. Compute and display RMSE, and R2 values for a data set.
2. Write a Python program to implement a logistic regression model.
3. Write a Python program to show the different pre-processing tasks such as Imputation, Standardization, Handling Categorical Variables, Outlier Management and Cross-Validation.
4. Write a Python program to implement the k-Nearest Neighbour algorithm to classify a data set.
5. Write a Python program to demonstrate how the decision tree algorithm works. Your program must consider an appropriate data set for building the decision tree and then apply the knowledge to classify a new sample.
6. Write a Python program to implement the naive Bayesian classifier for a sample training data set.
7. Write a Python program to perform classification using a Support Vector Machine (SVM).
8. Write a Python program to perform K means clustering on a data set.
9. Write a Python program to implement the random forest algorithm.
10. Write a Python program to implement the ADA BOOST algorithm.
11. Write a Python program to build an Artificial Neural Network (ANN) by implementing the Backpropagation algorithm and test the same using appropriate data sets.
12. Explore and Understand a Python code of any Deep Learning Model (CNN).

**Text Books:**

1. Tom M. Mitchell, "Machine Learning", Tata McGraw-Hill Education, 2017.
2. Ethem Alpaydin, "Introduction to Machine Learning", PHI, Third Edition, 2015.

**Reference Book(s):**

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", O'Reilly, Third Edition, 2022
2. Andreas Muller and Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'Reilly, 2016.

**Aim:**

The learning objectives of this course are to acquaint students with the theory and principles of Artificial Intelligence and Machine Learning and with representative practical systems.

**Course Outcomes (COs)**

**CO1:** Implement models based on supervised and unsupervised learning.

**CO3:** Evaluate and compare modelling aspects of various machine learning approaches using different evaluation measures.

**CO4:** Implement the ensemble methods for improving the learning algorithm.

**CO5:** Implement the concepts of neural networks and deep learning methods.

**No. of Hours: 24**

1. Write a Python program to implement a Linear Regression model. Compute and display RMSE, and R2 values for a data set.
2. Write a Python program to implement a logistic regression model.
3. Write a Python program to show the different pre-processing tasks such as Imputation, Standardization, Handling Categorical Variables, Outlier Management and Cross-Validation.
4. Write a Python program to implement the k-Nearest Neighbour algorithm to classify a data set.
5. Write a Python program to demonstrate how the decision tree algorithm works. Your program must consider an appropriate data set for building the decision tree and then apply the knowledge to classify a new sample.
6. Write a Python program to implement the naive Bayesian classifier for a sample training data set.
7. Write a Python program to perform classification using a Support Vector Machine (SVM).
8. Write a Python program to perform K means clustering on a data set.
9. Write a Python program to implement the random forest algorithm.
10. Write a Python program to implement the ADA BOOST algorithm.
11. Write a Python program to build an Artificial Neural Network (ANN) by implementing the Backpropagation algorithm and test the same using appropriate data sets.
12. Explore and Understand a Python code of any Deep Learning Model (CNN).

**Text Books:**

1. Tom M. Mitchell, "Machine Learning", Tata McGraw-Hill Education, 2017.
2. Ethem Alpaydin, "Introduction to Machine Learning", PHI, Third Edition, 2015.

**Reference Book(s):**

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", O'Reilly, Third Edition, 2022
2. Andreas Muller and Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'Reilly, 2016.

## School of Computer Science & Information Technology, DAVV, Indore

### Online Resources:

1. NPTEL Course Video Lectures on "Introduction to Machine Learning" - By Prof. Sudeshna Sarkar, IIT Kharagpur
2. Coursera Course Video Lectures on "Machine Learning" - By Prof. Andrew Ng, Stanford University

SCSIT, DAVV, Indore, MP - 452001, INDIA

## CS-6624: Network Management

### Aim:

To understand the principles of network management, different standards and protocols used in managing complex networks, the Automation of network management operations and making use of available network management systems.

---

### Course Outcomes:

- CO 1: Gain in-depth theoretical and practical knowledge of network management tools and the skill to use them in monitoring a network, in particular of SNMP (Simple Network Management Protocol).
  - CO 2: Analyze the challenges faced by Network managers
  - CO 3: Analyze and interpret the data provided by network management systems and take suitable actions.
  - CO 4: Apply the knowledge to manage organization's network management systems.
  - CO 5: -
- 

### Unit-I

Introduction: Computer Network, Goals and Applications, Data Communications and Network Management Overview: Communications protocols and Standards. Case Histories of Networking and Management, Network Management: Goals, Organization, and Functions, Network and System Management, Network Management System Platform, Current Status and future of Network Management.

Fundamentals of computer network technology: Network Topology, LANs, Network node components Hubs, Bridges, Routers, Gateways, Switches, WAN, ISDN Transmission Technology. Network Management Standards, Network Management Model, Organization Model, Information Model, Communication Model, ASN.1, Encoding Structure.

### Unit-II

SNMPv1 Network Management: Managed network: Case Histories and Examples, The History of SNMP Management. The SNMP Model, The Organization Model, System Overview. The Information Model, The SNMP Communication Model, Functional Model.

SNMPv2 Management: Major Changes in SNMPv2, SNMPv2 System Architecture, SNMPv2 Structure of Management Information. The SNMPv2 Management Information Base, SNMPv2 Protocol, Compatibility with SNMPv1.

### Unit-III

SNMP Management-RMON: Introduction, RMON SMI and MIB, RMON1.

Broadband Network Management-ATM Networks: Broadband Networks and Services, ATM Technology, ATM Network Management.

### Unit-IV

Broadband Network Management: Broadband Access Networks and Technologies, HFC Technology, HFC Management, DSL Technology, ADSL Technology, ADSL Management.

Telecommunication Management Network: Introduction, Operations Systems, TMN conceptual Model, TMN Architecture, TMN Management Service Architecture, An integrated view of TMN.

### **Unit-V**

Network Management Tools and Systems: Network Management Tools, Network Statistics Measurement Systems, History of Enterprise Management. Network Management systems, Commercial Network management Systems, System Management, Enterprise Management Solutions.

Network Management Applications: Configuration management, Fault management, performance management. Event Correlation Techniques, security Management, Accounting management, Report Management, Policy Based Management Service Level Management.

---

### **Text Books:**

Mani Subrahmanian, Network Management: Principles and Practice, Pearson Education, 2nd Edition 2010.

### **Reference Book(s):**

Morris, Network management, Pearson Education.

Mark Burges, Principles of Network System Administration, Wiley Dreamtech Publication.

Paul, Distributed Network Management, John Wiley Publication.

Andrew S. Tanenbaum, Computer Networks, Addison-Wesley, 4th Edition.

B.A. Forouzan, Data Communications and Networking, McGraw-Hill.