Jaya College of Arts and Science, Thiruninravur-602024.

Department of Computer Applications

Year : 2020-2021

Programme Offered :

- **M.C.A**

Programme Objective :

PO 1:	Computational Knowledge: Understand and apply mathematical foundation, computing and domain knowledge for the conceptualization of computing models from defined problems
PO 2:	Problem Analysis: Ability to identify, critically analyze and formulate complex computing problems using fundamentals of computer science and application domains
PO 3:	Design / Development of Solutions: Ability to transform complex business scenarios and contemporary issues into problems, investigate, understand and propose integrated solutions using emerging technologies
PO 4:	Conduct Investigations of Complex Computing Problems: Ability to devise and conduct experiments, interpret data and provide well informed conclusions
PO 5:	Modern Tool Usage: Ability to select modern computing tools, skills and techniques necessary for innovative software solutions
PO 6:	Professional Ethics: Ability to apply and commit professional ethics and cyber regulations in a global economic environment
PO 7:	Life-long Learning: Recognize the need for and develop the ability to engage in continuous learning as a Computing professional

PO 8:	Project Management and Finance: Ability to understand, management and computing principles with computing knowledge to manage projects in multidisciplinary environments
PO 9:	Communication Efficacy: Communicate effectively with the computing community as well as society by being able to comprehend effective documentations and presentations
PO 10:	Societal & Environmental Concern: Ability to recognize economical, environmental, social, health, legal, ethical issues involved in the use of computer technology and other consequential responsibilities relevant to professional practice
PO 11:	Individual & Team Work: Ability to work as a member or leader in diverse teams in multidisciplinary environment
PO 12:	Innovation and Entrepreneurship: Identify opportunities, entrepreneurship vision and use of innovative ideas to create value and wealth for the betterment of the individual and society

Programme Outcome :

Programme Specific Outcomes	
PSO 1:	Ability to pursue careers in IT industry/ consultancy/ research and development, teaching and allied areas related to computer science.
PSO 2:	Comprehend, explore and build up computer programs in the areas allied to Algorithms, System Software, Multimedia, Web Design and Big Data Analytics for efficient design of computer-based systems of varying complexity.

First Semester

Course components	Name of Course	Credits	Max.Marks		TOTAL
			CIA	UE	
Core– 1	C++ & Data Structures	4	25	75	100
Core- 2	Digital Computer Fundamentals	4	25	75	100
Core- 3	Database Management Systems	4	25	75	100
Core- 4	Practical–I: Data Structures using C++ Lab.	2	40	60	100
Core- 5	Practical–II: RDBMS Lab.	2	40	60	100
Extra – Disciplinary- I	Accounting & Financial Management	3	25	75	100
Elective I	Choose any one	3	25	75	100
Soft Skill-1	Choose any one	2	40	60	100

Second Semester

Course components	Name of Course	Credits	Max. Marks		TOTAL
			CIA	UE	
Core – 6	Design and Analysis of Algorithms	4	25	75	100
Core – 7	Object Oriented Analysis and Design	4	25	75	100
Core – 8	Artificial Intelligence	4	25	75	100
Elective II	Choose any one	3	25	75	100
Core – 9	Practical – III: Object Oriented Analysis and Design Lab	2	40	60	100
Core – 10	Practical – IV: Web Based Application Development Lab.	2	40	60	100
Extra– Disciplinary - II	Web Based Application Development	3	25	75	100
Soft Skill-2	Choose any one	2	40	60	100
Soft Skill-3	Choose any one	2	40	60	100
Internship	During I year summer vacation 4 to 6 weeks – Evaluation will be at the end of third semester.				

Third Semester

Course components	Name of Course	Credits	Max. Marks		TOTAL
			CIA	UE	
Core – 11	Machine Learning	4	25	75	100
Elective III	Choose any one	3	25	75	100
Elective IV	Choose any one	3	25	75	100
Elective V	Choose any one	3	25	75	100
Core– 12	Practical – V: Machine Learning Lab	2	40	60	100
Core -13	Practical – VI:Group Project	2	40	60	100
Soft Skill-4	Choose any one	2	40	60	100
Internship	During I year summer vacation 4 to 6 weeks	2			100

Fourth Semester

Course components	Name of Course	Credits	Max. Marks		TOTAL
			CIA	UE	
Core-14	Project & viva-voce	20	20	60+20	100

* CIA = Continuous Internal Assessment, UE = University Examination

Course Name: C++ and Data Structures

Course Objective:

- CO1: The basic programming and OOPs concepts
- CO2: Creating C++ programs
- CO3: Tokens, expressions and control structures in C++
- CO4: Arranging same data systematically with arrays
- CO5: Be familiar with basic techniques of algorithm analysis
- CO5: Be familiar with writing recursive methods
- CO6: Master the implementation of linked data structures such as linked lists and binary trees

Course OutCome:

- CO1: Classify different data structures such as stack, queues, linked list, trees and graphs
- CO2: Analyze and implement various searching and sorting techniques
- CO3: Describe OOPs concepts
- CO4: Use functions and pointers in your C++ program
- CO5 : Understand tokens, expressions, and control structures

Syllabus:

Unit 1:
Introduction to C++; Tokens, Keywords, Identifiers, Variables,Operators, Manipulators, Expressions andControlStructuresinC++; Pointers-FunctionsinC++- MainFunction-FunctionPrototyping-ParametersPassinginFunctions-ValuesReturnbyFunctions- InlineFunctions-FriendandVirtualFunctions
Unit-2:
ClassesandObjects;ConstructorsandDestructors;andOperatorOverloadingandType Conversions - Type of Constructors -Function overloading. Inheritance:SingleInheritance-MultilevelInheritance- MultipleInheritance-HierarchicalInheritance- HybridInheritance. Pointers, Virtual Functions and Polymorphism;Managing Console I/Ooperations.
Unit3:
WorkingwithFiles: ClassesforFileStreamOperations- OpeningandClosingaFile -End-of-File Deduction -File Pointers -Updating aFile -Error Handling duringFile Operations -Command-line Arguments. DataStructures: Definition ofaDatastructure-primitiveandcompositeDataTypes,Asymptoticnotations,Arrays,OperationsonArrays,Orderlists.
Unit-4:
Stacks - Applications of Stack - Infix to Postfix Conversion, Recursion,Maze Problems - Queues- Operations onQueues, QueueApplications, Circular Queue.Singly Linked List-Operations, Application - Representation of a Polynomial,PolynomialAddition;DoublyLinkedList-Operations,Applications.
Unit-5:
TreesandGraphs:BinaryTrees-ConversionofForesttoBinaryTree,Operations- Tree Traversals; Graph - Definition, Types of Graphs, Hashing Tables andHashing Functions,Traversal-ShortestPath;Dijkstra'sAlgorithm.

Course name: Digital Computer Fundamentals

Course objective:

- CO1:Give students an in-depth understanding of why computers are essential components in business, education and society.
- CO2: -Introduce the fundamentals of computing devices and reinforce computer vocabulary, particularly with
- CO3: respect to personal use of computer hardware and software, the Internet, networking and mobile computing.
- CO4: -Provide hands-on use of Microsoft Office 2013 applications Word, Excel, Access and PowerPoint.
- CO5: Completion of the assignments will result in MS Office applications knowledge and skills.

Course OutCome:

- CO1:Describe the usage of computers and why computers are essential components in business andsociety.
- CO2: Utilize the Internet Web resources and evaluate on-line e-business system.
- CO3: Solve common business problems using appropriate Information Technology applications andsystems.
- CO4:Identify categories of programs, system software and applications. Organize and work with files and folders.
- CO5: Describe various types of networks network standards and communication software.

Syllabus:

Unit 1 :

Number System – Converting numbers from one base to another – Complements – Binary Codes – Integrated Circuits – Boolean algebra – Properties of Boolean algebra – Boolean functions – Canonical and Standard forms – Logical Operations – Logic gates – Karnaugh Map up to 6 variables – Don’t Care Condition – Sum of Products and Products of Sum simplification – Tabulation Method.

Unit-2 :

Adder – Subtractor – Code Converter – Analyzing a combinational Circuit – Multilevel NAND and NOR circuits – Properties of XOR and equivalence functions – Binary Parallel Adder – Decimal Adder – Magnitude Comparator – Decoders – Multiplexers – ROM – PLA.

Unit 3 :

Flip Flops – Triggering of flip-flops – Analyzing a sequential circuit – State reduction – excitation tables – Design of sequential circuits – Counters – Design with state equation – Registers – Shift Registers – Ripple and synchronous Counters.

Unit-4 :

Memory Unit – Processor Organization - Bus Organization – Scratch Pad memory – ALU – Design of ALU – Status Register – Effects of Output carry – Design of Shifter – Processor Unit – Microprogramming – Design of specific Arithmetic Circuits

Unit-5 :

Accumulator – Design of Accumulator – Computer Design – System of Configuration – Instruction and Data formats – Instruction sets – Timing and Control – Execution of Instruction – Design of Computer – Hardwired control – PLA Control and Microprogram control

Course Name: Database Management Systems

Course Objective:

- CO1:Knowledge of DBMS, both in terms of use and implementation/design
- CO2: Experience with SQL
- CO3: Increased proficiency with the programming language C++
- CO4: Experience working as part of team
- CO5: Experience with analysis and design of (DB) software

Course OutCome:

- CO1: Describe the fundamental elements of relational database management systems
- CO2: Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
- CO3: Design ER-models to represent simple database application scenarios
- CO4: Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
- CO5: Improve the database design by normalization.

Syllabus:

Unit 1: Introduction to Database Systems – Relational Model – Structure – Relational Algebra – Null Values – SQL – Set Operation – Views – Advanced SQL – Embedded SQL – Recursive Queries – The Tuple Relational Calculus – Domain Relational Calculus.

Unit 2: E-R Model – Constraints – E-R- Diagrams Weak Entity Sets – Reduction to Relational Schemes – Relational Database Design – Features of Relational Design – Automatic Domains and First Normal Form – Decomposition using Functional Dependencies – Multivalued Dependencies – More Normal Forms – Web Interface – Object – Based Databases – Structured Types and inheritance in SQL – Table inheritance – Persistent.

Unit 3: Storage and File Structure – RAID – File Organization – Indexing and Hashing – B Tree – B Tree Index files - Static and Dynamic Hashing – Query Processing – Sorting & Join Operators – Query Optimization – Choice of Evaluation Plans.

Unit 4: Transaction Management – Implementation of Atomicity and Durability – Serializability – Recoverability – Concurrency Control – Dead Lock Handling – Recovery System – Buffer Management.

Unit 5 : Database – System Architecture – Client Server – Architectures – Parallel System – Network Types – Distributed Database – Homogeneous and Hetrogeneous Database – Directory System – Case Study

Course Name: Data Structures using C++ Lab

Course Objective:

- CO1: To provide the knowledge of basic data structures and their implementations.
- CO2: . To understand importance of data structures in context of writing efficient programs.
- CO3: To develop skills to apply appropriate data structures in problem solving.

Course OutCome:

- CO1: Be able to design and analyze the time and space efficiency of the data structure
- CO2: Be capable to identity the appropriate data structure for given problem
- CO3: Have practical knowledge on the applications of data structures

Syllabus:

Fortheimplementationofthefollowingproblems,thestudentsareadvisedtouseallpossibleobjectorientedfeatures. The implementation based on structured concepts will not accepted.

1. Implementation of Arrays(Single and Multi-Dimensional)
2. Polynomial Object and necessary overloaded operators.
3. Singly Linked Lists.
4. Circular Linked Lists.
5. Doubly Linked Lists.
6. Implementation of Stack(using Arrays and Pointers)
7. Implementation of Queue(Using Arrays and Pointers)
8. Implementation of Circular Queue(using Arrays and Pointers)
9. Evaluation of Expressions.
10. Binary Tree implementations and Traversals.
11. Binary Search Trees.

Course Name: Practical – II: RDBMS Lab

Course Objective:

- CO1: to provide a sound introduction to the discipline of database management as a subject in its own right, rather than as a compendium of techniques and product-specific tools.
- CO2: to familiarize the participant with the nuances of database environments towards an information-oriented data-processing oriented framework
- CO3: to give a good formal foundation on the relational model of data
- CO4: to present SQL and procedural interfaces to SQL comprehensively
- CO5: to give an introduction to systematic database design approaches covering conceptual design, logical design and an overview of physical design

Course OutCome:

- CO1: Understand, appreciate and effectively explain the underlying concepts of database technologies
- CO2: Design and implement a database schema for a given problem-domain
- CO3: Normalize a database
- CO4: Populate and query a database using SQL DML/DDL commands.
- CO5: Declare and enforce integrity constraints on a database using a state-of-the-artRDBMS
- CO6: Programming PL/SQL including stored procedures, stored functions, cursors,packages.
- CO7: Design and build a GUI application using a 4GL

SYLLABUS:

1. Library Information Processing.
2. Students Mark sheet processing.
3. Telephone directory maintenance.
4. Gas booking and delivery system.
5. Electricity Bill Processing.
6. Bank Transactions (SB).
7. Pay roll processing.
8. Inventory
9. Question Database and conducting quiz.
10. Purchase order processing.

I Year & II Semester

Course Name: Design and Analysis of Algorithms

Course Objective:

- CO1:Analyze the asymptotic performance of algorithms.
- CO2:Write rigorous correctness proofs for algorithms
- CO3:Demonstrate a familiarity with major algorithms and data structures.
- CO4:Apply important algorithmic design paradigms and methods of analysis.
- CO5:Synthesize efficient algorithms in common engineering design situations.

Course OutCome:

- CO1 Argue the correctness of algorithms using inductive proofs and invariants.
- CO2:Analyze worst-case running times of algorithms using asymptotic analysis.
- CO3:Describe the divide-and-conquer paradigm and explain when an algorithmic design
- CO4:situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-CO5:conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.
- CO6: Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.

Syllabus:

Unit 1: Introduction - Definition of Algorithm – pseudocode conventions – recursive algorithms – time and space complexity –big-“oh” notation – practical complexities – randomized algorithms – repeated element – primality testing **-** Divide and Conquer:General Method - Finding maximum and minimum – merge sort.

Unit-2: Divide and conquer contd. – Quicksort, Selection, Strassen's matrix multiplication – Greedy Method:General Method –knapsack problem - Tree vertex splitting - Job sequencing with dead lines – optimal storage on tapes.

Unit 3: Dynamic Programming: General Method - multistage graphs – all pairs shortest paths – single source shortest paths - String Editing – 0/1 knapsack.Search techniques for graphs – DFS-BFS-connected components – biconnected components.

Unit 4: Back Tracking: General Method – 8-queens - Sum of subsets - Graph Coloring – Hamiltonian cycles. Branch and Bound: General Method - Traveling Salesperson problem.

Unit 5: Lower Bound Theory:Comparison trees - Oracles and advisory arguments - Lower bounds through reduction - Basic Concepts of NP-Hard and NP-Complete problems.

Course Name: Object Oriented Analysis and Design

Course Objective:

- CO1:To understand the Object-based view of Systems
- CO2:To develop robust object-based models for Systems
- CO3:To inculcate necessary skills to handle complexity in software design
- CO4:To learn the basis of OO Analysis and design
- CO5:To have clear idea about traditional and modern SW development Methodologies and OOPS concepts.
- CO6: To identify objects, relationships, services and attributes
- CO7: To introduce the concept of Object-oriented system development lifecycle
- CO8:Discuss the overview of Object oriented methodologies

Course OutCome:

- CO1. Analyse, design, document the requirements through use case driven approach.
- CO2.Identify, analyse, and model structural and behavioural concepts of the system.
- CO3.Develop,explore the conceptual model into various scenarios and applications.
- CO4.Apply the concepts of architectural design for deploying the code for software.

Syllabus

Unit 1: System Development - Object Basics - Development Life Cycle - Methodologies - Patterns - Frameworks - Unified Approach - UML.

Unit-2: Use-Case Models - Object Analysis - Object relations - Attributes - Methods – Class and Object responsibilities - Case Studies.

Unit 3: Design Processes - Design Axioms - Class Design - Object Storage - Object Interoperability - Case Studies.

Unit-4: User Interface Design - View layer Classes - Micro-Level Processes - View Layer Interface - Case Studies.

Unit-5: Quality Assurance Tests - Testing Strategies - Object orientation on testing - Test Cases - test Plans - Continuous testing - Debugging Principles - System Usability - Measuring User Satisfaction - Case Studies.

Course Name: Artificial Intelligence

Course Objective:

- CO1:The aim of Artificial Intelligence & Machine Learning course is to prepare students for career in computer science & engineering where knowledge of AI & ML techniques leading to the advancement of research and technology
- CO2:Artificial Intelligence and Machine Learning are the terms of computer science.
- CO3:Machine Learning is the learning in which machine an learn by its own without being explicitly programmed.
- CO4:It is an application of AI that provide system the ability to automatically learn and improve from experience

Course OutCome:

- CO1: Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
- CO2:Demonstrate proficiency in applying scientific method to models of machine learning.
- CO3:Discuss the awareness of ANN and different optimizations techniques
- CO4: Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
- CO5: Demonstrate proficiency in applying scientific method to models of machine learning.
- CO6:Discuss the basics of ANN and different optimizations techniques.

Syllabus:

UNIT I:
Introduction: What Is AI? - Foundations of Artificial Intelligence-The History of Artificial Intelligence-The State of the Art- Risks and Benefits of AI. Intelligent Agents: Agents and Environments - The Concept of Rationality - The Nature of Environments- The Structure of Agents.

UNIT II:
Solving problem by Searching: Problem-Solving Agents - Example Problems - Search Algorithms: Best-first search - Search data structures - Redundant paths - Measuring problem-solving performance - Uninformed Search Strategies: BFS-DFS- Depth limited and iterative deepening search. Heuristic Search Strategies: Greedy best-first search - A* search - Search contours - Inadmissible heuristics and weighted A* - Heuristic Functions.

UNIT III:
Local Search and Optimization Problems:Hill-climbing search - Simulated annealing - Local beam search - Local Search in Continuous Spaces - Search with Nondeterministic Actions: The erratic vacuum world - AND—OR search trees. Optimal Decisions in Games: The minimax search algorithm - Optimal decisions in multiplayer games - Alpha--Beta Pruning. Heuristic Alpha--Beta Tree Search: Evaluation functions - Cutting off search - Forward pruning - Monte Carlo Tree Search - Stochastic Games-Limitations of Game Search Algorithms.

UNIT IV:
Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems - Constraint Propagation: Inference in CSPs - Backtracking Search for CSPs - Local Search for CSPs - The Structure of Problems. Logical agent and Logics: Propositional Logic - Propositional Theorem Proving - Effective Propositional Model Checking - Agents Based on Propositional Logic - First-Order Logic: Syntax and Semantics of First-Order Logic - Using First-Order Logic - Knowledge Engineering in First-Order Logic. Inference in

First-Order Logic: Unification and First-Order Inference - Forward Chaining - Backward Chaining – Resolution.

UNIT V:

Knowledge Representation and Reasoning : Ontological Engineering - Categories and Objects - Events - Mental Objects and Modal Logic - Reasoning Systems for Categories - Reasoning with Default Information. Automated Planning: Definition of Classical Planning - Algorithms for Classical Planning - Heuristics for Planning. Quantifying Uncertainty: Acting under Uncertainty - Basic Probability Notation - Inference Using Full Joint Distributions - Independence - Bayes' Rule and Its Use - Naive Bayes Models

Course Name: Web Based Application Development
Course Objective:

- CO1: Understand the principles of creating an effective web page, including an in-depth consideration of information architecture.
- CO2:Become familiar with graphic design principles that relate to web design and learn how to implement theories into practice.
- CO3:Develop skills in analyzing the usability of a web site.
- CO4:Understand how to plan and conduct user research related to web usability.
- CO6: Learn the language of the web: HTML and CSS.
- CO6: Learn CSS grid layout and flexbox.

Course OutCome:

- CO1: Students will be able to write a well formed / valid XML document.
- CO2:Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
- CO3:Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
- CO7:Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database

Syllabus:

Unit – I: OVERVIEW OF ASP.NET - The .NET framework – The C# Language: Data types – Declaring variables- Scope and Accessibility- Variable operations- Object Based manipulation- Conditional Structures- Loop Structures- Methods. Types, Objects and Namespaces : The Basics about Classes- Value types and Reference types- Understanding name spaces and assemblies - Advanced class programming. .
Unit – II: Developing ASP.NET Applications - The Anatomy of a Web Form – Writing Code - Visual Studio Debugging. Web Form Fundamentals: The Anatomy of an ASP.NET Application - Introducing Server Controls - HTML Control Classes - The Page Class - Application Events - ASP.NET Configuration. Web Controls: Web Control Classes - List Controls - Web Control Events and AutoPostBack - A Simple Web Page.

Unit – III: Error Handling, Logging, and Tracing: Common Errors - Exception Handling - Handling Exceptions - Throwing Your Own Exceptions - Logging Exceptions - Page Tracing. State Management: View State - Transferring Information Between Pages – Cookies - Session State - Session State Configuration - Application State. Validation: Understanding Validation - The Validation Controls.

Unit – IV: Rich Controls: The Calendar - The AdRotator - Pages with Multiple Views - User Controls and Graphics - User Controls - Dynamic Graphics . Website Navigation: Site Maps - URL Mapping and Routing - The SiteMapPath Control - The TreeView Control - The Menu Control. ADO.NET Fundamentals: The Data Provider Model - Direct Data Access - Disconnected Data Access.

Unit – V: Data Binding: Single-Value Data Binding - Repeated-Value Data Binding - Data Source Controls - The Data Controls: The GridView - Formatting the GridView - Selecting a GridView Row - Editing with the GridView - Sorting and Paging the GridView - Using GridView Templates - The DetailsView and FormView – XML: The XML Classes - XML Validation - XML Display and Transforms. Website Security: Security Fundamentals - Understanding Security - Authentication and Authorization - Forms Authentication - Windows Authentication.

Course Name: Object Oriented Analysis and Design Lab

Course Objective:

- CO1 : Specify, analyze and design the use case driven requirements for a particular system.
- CO2:Model the event driven state of object and transform them into implementation specific layouts.
- CO3:Identify, Analyze the subsystems, various components and collaborate them interchangeably.

Course OutCome:

- CO1. Analyse, design, document the requirements through use case driven approach.
- CO2.Identify, analyse, and model structural and behavioural concepts of the system.
- CO3.Develop,explore the conceptual model into various scenarios and applications.
- CO4.Apply the concepts of architectural design for deploying the code for software.

Syllabus

Software: Rational Rose or Any **UML case Tools**

OBJECTIVE: To Design UML and Use case diagram for given scenario/domain using OOAD methodologies by capturing the key functional requirements listed below:

1. Identify Use Cases and develop the Use Case model.
2. Identify the business activities and develop an UML Activity diagram.
3. Identity the conceptual classes and develop a domain model with UML Class diagram.
4. Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.
5. Draw the State Chart diagram.
6. Identify the User Interface, Domain objects, and Technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.
7. Implement the Technical services layer.
8. Implement the Domain objects layer.
9. Implement the User Interface layer.
10. Draw Component and Deployment diagrams.

Suggested domains for Practical/Lab Experiment

1. Student information system.
2. Stock Maintenance System.
3. Banking system.
4. Online course reservation system.
5. Exam Registration.
6. Employee Management System.
7. Project Tracking System.
8. Library Information System.
9. E-ticketing
10. E-book management system.
11. Recruitment system.
12. Conference Management System.
13. BPO Management System.
14. Credit card processing.

Course Name: Web Based Application Development Lab

Course Objective:

- CO1: Design and implement dynamic websites with good aesthetic sense of designing and latest technical know-how's.
- CO2: Have a Good grounding of Web Application Terminologies, Internet Tools, E – Commerce and other web services.
- CO3: Get introduced in the area of Online Game programming.
- CO4: Understand the principles of creating an effective web page, including an in-depth consideration of information architecture.
- CO5: Become familiar with graphic design principles that relate to web design and learn how to implement theories into practice.
- CO6: Develop skills in analyzing the usability of a web site.
- CO7: Understand how to plan and conduct user research related to web usability.
- CO8:Learn the language of the web: HTML and CSS.

Course OutCome:

- CO1: Comprehend and propose Web Application infrastructure.
- CO2: Apply client/server communication techniques such as server, application, session variables, cookies and server behaviours.
- CO3: Determine the needs for web database and connectivity.
- CO4: Apply code reuse with templates, libraries, and snippets.
- CO5: Evaluate several alternatives in the design of a web application.
- CO6:Develop a functional web application

Syllabus:

1. Create a Multilevel inheritance for Employee using appropriate data members and methods using C#.
2. Create an application form to apply for a new course in a college, fill the information and submit it (Use Basic Web Server controls).
3. Create a web application using Global.asax file which will count the number of visitors on web page.
4. Design a web page to implement upload and download files functionality using File Upload Control.
5. Develop a web page to implement the concept of state management using Cookies
6. Develop a web page to implement the concept of state management using Session and Application
7. Develop a web page to implement the concept of state management using ViewState and QueryString.
8. Design Sign Up form and validate User Name (Minimum 8 character Maximum 15 and only characters and underscore), Password (Minimum 8 Characters) and Confirm_Password (Both should be same), Phone No (Only digits), Email-id etc. (Use Validation controls).
9. Create a web site using Master Page Concept having two content pages.
10. Write sample application to connect to database, Fetching and inserting data from database and using Data Reader
11. Create Employee database and develop a web application to Add, Update, View and Delete records from database using in Gridview control.
12. Create Student database and develop a web application to Add, Update, View and Delete records from database using in Gridview control.
13. Create a web form for Online Library data entry and manipulate records using C#.NET.
14. Design a web page to display the XML content.
15. Design a web application to demonstrate form authentication and authorization.

II Year & III Semester

Course Name: Machine Learning

Course Objective:

- CO1:The aim of Artificial Intelligence & Machine Learning course is to prepare students for career in computer science & engineering where knowledge of AI & ML techniques leading to the advancement of research and technology.
- CO2: Artificial Intelligence and Machine Learning are the terms of computer science. Machine Learning is the learning in which machine an learn by its own without being explicitly programmed .
- CO3: It is an application of AI that provide system the ability to automatically learn and improve from experience.

Course OutCome:

- CO1 Demonstrate fundamental understanding of artificial intelligence (AI) and expertsystems.
- CO2:Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
- CO3 Demonstrate proficiency in applying scientific method to models of machine learning.
- CO4:Discuss the awareness of ANN and different optimizations techniques
- CO5: Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
- CO6:Demonstrate proficiency in applying scientific method to models of machine learning.
- CO7:Discuss the basics of ANN and different optimizations techniques

Syllabus

Unit 1: The Fundamentals of Machine Learning: The Machine Learning Landscape - Types of Machine Learning Systems - Main Challenges of Machine Learning - Testing and Validating. End-to-End Machine Learning Project - 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 - Fine-Tune Your Model - Launch, Monitor, and Maintain Your System.

Unit 2: Ingredients of machine learning: Tasks – Models – Features. Supervised Learning: Classification – Binary classification and related tasks – Scoring and ranking – class probability estimation – Multi-class classification. Unsupervised Learning: Regression – Unsupervised and descriptive learning. Concept Learning: The hypothesis space – paths through the hypothesis space – beyond conjunctive concepts – learnability.

Unit 3: Tree Models: Decision trees – Ranking and probability estimation trees – tree learning as variance reduction. Rule Models: Learning ordered rule lists – learning unordered rule sets – descriptive rule learning – first–order rule learning. Linear Models: The least-squares method – The perceptron – Support vector machines.

Unit 4: Distance-based Models: Neighbours and exemplars – Nearest-neighbour classification – Distance-based clustering – K-Means algorithm – Hierarchical clustering. Probabilistic Models: The normal distribution and its geometric interpretations – probabilistic models for categorical data – Naïve Bayes model for classification – probabilistic models with hidden values – Expectation-Maximization.

Unit 5: Features: Kinds of features – Feature transformations – Feature construction and selection. Model ensembles: Bagging and random forests – Boosting – Mapping the ensemble landscape. Machine Learning experiments: What to measure – How to measure it – How to interpret it.

Course Name: : Machine Learning Lab

Course Objective:

- CO1: To understand the basic theory underlying machine learning.
- CO2: To be able to formulate machine learning problems corresponding to different applications.
- CO3: To understand a range of machine learning algorithms along with their strengths and weaknesses.
- CO4: To be able to apply machine learning algorithms to solve problems of moderate complexity.
- CO5:To apply the algorithms to a real-world problem, optimize the models learned and report on the expected
- CO6: accuracy that can be achieved by applying the models.

Course OutCome:

- *Co1:Design* and evaluate the unsupervised models through python in built functions.
- Co2:Evaluate the machine learning models pre-processed through various feature engineering algorithms by python programming.
- *Co3:Design* and *apply* various reinforcement algorithms to solve real time complex problems.
- *Co4:Design* and *develop* the code for recommender system using Natural Language processing
- *Con5:Understand* the basic concepts of deep neural network model and design the same.

Syllabus:

Machine Learning Tools and Applications:

Machine learning platform: WEKA machine learning workbench, R platform, Python Scipy.
Machine Learning Library: scikit-learn in Python, JSAT in Java, Accord Framework in .NET
GUIs: KNIME, RapidMiner, Orange.
Applications: Prediction using data, Speech recognition, Healthcare, Object recognition in images, Natural Language Processing, Online search

1. Data Preprocessing:
    a. Data Cleaning
    b. Data Transformation
    c. Data Reduction
    d. Feature extraction
2. Supervised learning:
    a. Decision tree classification
    b. Classification using Support Vector Machines
    c. Classification using Multilayer perceptron
3. Unsupervised learning:
    a. Regression
    b. K-Means clustering
    c. Hierarchical clustering

Mini Project: Application of Data Preprocessing techniques and Machine Learning techniques on a data set selected from UCI repository / Kaggle / Government and submission of a report.