

CONTENTS

- i. PO and PSO Description
- ii. PG – Template
- iii. Methods of Evaluation & Methods of Assessment
- iv. Semester Index.
- v. Subjects – Core, Elective, Non-major, Skill Enhanced, Ability Enhanced, Extension Activity, Environment, Professional Competency
 - 1) Course Lesson Box
 - 2) Course Objectives
 - 3) Units
 - 4) Learning Outcome
 - 5) Reference and Text Books
 - 6) Web Sources
 - 7) PO & PSO Mapping tables

TANSCHÉ REGULATIONS ON LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK FOR POSTGRADUATE EDUCATION	
Programme	M.C.A.,
Programme Code	
Duration	PG - Two Years
Programme Outcomes (Pos)	<p>PO1: Problem Solving Skill Apply knowledge of Management theories and Human Resource practices to solve business problems through research in Global context.</p> <p>PO2: Decision Making Skill Foster analytical and critical thinking abilities for data-based decision-making.</p> <p>PO3: Ethical Value Ability to incorporate quality, ethical and legal value-based perspectives to all organizational activities.</p> <p>PO4: Communication Skill Ability to develop communication, managerial and interpersonal skills.</p> <p>PO5: Individual and Team Leadership Skill Capability to lead themselves and the team to achieve organizational goals.</p> <p>PO6: Employability Skill Inculcate contemporary business practices to enhance employability skills in the competitive environment.</p> <p>PO7: Entrepreneurial Skill Equip with skills and competencies to become an entrepreneur.</p> <p>PO8: Contribution to Society Succeed in career endeavors and contribute significantly to society.</p> <p>PO 9 Multicultural competence Possess knowledge of the values and beliefs of multiple cultures and a global perspective.</p> <p>PO 10: Moral and ethical awareness/reasoning Ability to embrace moral/ethical values in conducting one's life.</p>
Programme Specific Outcomes (PSOs)	<p>PSO1 – Placement To prepare the students who will demonstrate respectful engagement with others' ideas, behaviors, beliefs and apply diverse frames of reference to decisions and actions.</p>

	<p>PSO 2 - Entrepreneur To create effective entrepreneurs by enhancing their critical thinking, problem solving, decision making and leadership skill that will facilitate startups and high potential organizations.</p> <p>PSO3 – Research and Development Design and implement HR systems and practices grounded in research that comply with employment laws, leading the organization towards growth and development.</p> <p>PSO4 – Contribution to Business World To produce employable, ethical and innovative professionals to sustain in the dynamic business world.</p> <p>PSO 5 – Contribution to the Society To contribute to the development of the society by collaborating with stakeholders for mutual benefit.</p>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Choice Based Credit System (CBCS), Learning Outcomes Based Curriculum Framework (LOCF) Guideline Based Credits and Hours Distribution System
for all Post – Graduate Courses including Lab Hours**

First Year – Semester – I

Part	List of Courses	Credits	No. of Hours
	Core – I	5	7
	Core – II	5	7
	Core – III	4	6
	Elective – I	3	5
	Elective – II	3	5
		20	30

Semester-II

Part	List of Courses	Credits	No. of Hours
	Core – IV	5	6
	Core – V	5	6
	Core – VI	4	6
	Elective – III	3	4
	Elective – IV	3	4
	Skill Enhancement Course [SEC] – I	2	4
		22	30

Second Year – Semester – III

Part	List of Courses	Credits	No. of Hours
	Core – VII	5	6
	Core – VIII	5	6
	Core – IX	5	6
	Core (Industry Module) – X	4	6
	Elective – V	3	3
	Skill Enhancement Course – II	2	3
	Internship / Industrial Activity [Credits]	2	-
		26	30

Semester-IV

Part	List of Courses	Credits	No. of Hours
	Core – XI	5	6
	Core – XII	5	6
	Project with VIVA VOCE	7	10
	Elective – VI (Industry Entrepreneurship)	3	4
	Skill Enhancement Course – III / Professional Competency Skill	2	4
	Extension Activity	1	-
		23	30

Total 91 Credits for PG Courses

METHODS OF EVALUATION		
Internal Evaluation	Continuous Internal Assessment Test	25 Marks
	Assignments / Snap Test / Quiz	
	Seminars	
	Attendance and Class Participation	
External Evaluation	End Semester Examination	75 Marks
Total		100 Marks
METHODS OF ASSESSMENT		
Remembering (K1)	<ul style="list-style-type: none"> • The lowest level of questions require students to recall information from the course content • Knowledge questions usually require students to identify information in the textbook. 	
Understanding (K2)	<ul style="list-style-type: none"> • Understanding of facts and ideas by comprehending, organizing, comparing, translating, interpolating and interpreting in their own words. • The questions go beyond simple recall and require students to combine data together 	
Application (K3)	<ul style="list-style-type: none"> • Students have to solve problems by using/applying a concept learned in the classroom. • Students must use their knowledge to determine an exact response. 	
Analyze (K4)	<ul style="list-style-type: none"> • Analyzing the question is one that asks the students to breakdown something into its component parts. • Analyzing requires students to identify reasons, causes or motives and reach conclusions or generalizations. 	
Evaluate (K5)	<ul style="list-style-type: none"> • Evaluation requires an individual to make judgment on something. • Questions to be asked to judge the value of an idea, a character, a work of art, or a solution to a problem. • Students are engaged in decision-making and problem-solving. • Evaluation questions do not have single right answers. 	
Create (K6)	<ul style="list-style-type: none"> • The questions of this category challenge students to get engaged in creative and original thinking. • Developing original ideas and problem solving skills 	

**PROGRAMME OUTCOMES (PO) - PROGRAMME SPECIFIC OUTCOMES (PSO)
MAPPING**

PROGRAMME SPECIFIC OUTCOMES (PSO)					
	PO1	PO2	PO3	PO4	PO5
PSO1	3	3	3	3	3
PSO2	3	3	3	3	3
PSO3	3	3	3	3	3
PSO4	3	3	3	3	3
PSO5	3	3	3	3	3

Level of Correlation between PO's and PSO's

(Suggested by UGC as per Six Sigma Tool – Cause and Effect Matrix)

Assign the value

1 – Low

2 – Medium

3 – High

0 – No Correlation

MASTER OF COMPUTER APPLICATIONS (MCA)

Course	Number of Credits	Hours Per Week	Examination Duration (hrs)	Marks		
				I. A	ESE	Total
Semester – I						
Core – I Theory Discrete Mathematics	5	7	3	25	75	100
Core – II Theory Linux and shell programming	5	7	3	25	75	100
Core – III Theory Python Programming	4	6	3	25	75	100
Elective – I Data Engineering and Management / Data Engineering and Management Lab / Architecture and Frameworks / Architecture and Frameworks Lab / Linux and Shell Programming Lab	3	5	3	40	60	100
Elective – II Software Development Technologies / Software Development Technologies Lab / Soft Computing / Soft Computing Lab / Python Programming Lab	3	5	3	40	60	100
	20	30				

Discrete Mathematics

Course Objective

- To know the concepts of relations and functions.
- To distinguish among different normal forms and quantifiers.
- To solve recurrence relations and permutations & combinations.
- To know and solve matrices, rank of matrix & characteristic equations.
- To study the graphs and its types.

Unit-I

Relations- Binary relations-Operations on relations- properties of binary relations in a set – Equivalence relations-- Representation of a relation by a matrix -Representation of a relation by a digraph – **Functions**-Definition and examples-Classification of functions-Composition of functions-Inverse function.

Unit-II

Mathematical Logic-Logical connectives-**Well formed formulas** – Truth table of well formed formula –Algebra of proposition –Quine’s method-**Normal forms of well formed formulas**- Disjunctive normal form-Principal Disjunctive normal form-Conjunctive normal form-Principal conjunctive normal form-**Rules of Inference for propositional calculus** – **Quantifiers**- Universal Quantifiers- Existential Quantifiers.

Unit-III

Recurrence Relations- Formulation -solving recurrence Relation by Iteration- solving Recurrence Relations- Solving Linear Homogeneous Recurrence Relations of Order Two- Solving Linear Non homogeneous Recurrence Relations. **Permutations**-Cyclic permutation- Permutations with repetitions- permutations of sets with indistinguishable objects-**Combinations**- Combinations with repetition.

Unit-IV

Matrices- special types of matrices-Determinants-Inverse of a square matrix-Cramer’s rule for solving linear equations-Elementary operations-Rank of a matrix-solving a system of linear equations-characteristic roots and characteristic vectors-Cayley-Hamilton Theorem-problems.

Unit-V

Graphs -Connected Graphs -Euler Graphs- Euler line-Hamiltonian circuits and paths -planar graphs – Complete graph-Bipartite graph-Hyper cube graph-Matrix representation of graphs

Text book

1. N.Chandrasekaran and M.Umaparvathi, Discrete mathematics, PHI Learning Private Limited, New Delhi, 2010.

Reference Book

1. Kimmo Eriksson &Hillevi Gavel, Discrete Mathematics & Discrete Models, Studentlitteratur AB, 2015.
2. Kenneth H. Rosen Discrete Mathematics and applications, Mc Graw Hill, 2012

Course Outcomes

On the successful completion of the course, students will be able

CO1:	To understand the concepts of relations and functions distinguish among normal forms	K2	IO
CO2:	To analyze and evaluate the recurrence relations	K4,K5	HO
CO3:	To distinguish among various normal forms and predicate calculus	K5	H O
CO4:	To solve and know various types of matrices	K1	L O
CO5:	To evaluate and solve various types of graphs	K5	H O

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Linux and Shell Programming

Course Objective

- To teach principles of operating system including File handling utilities, Basic Linux commands, Scripts and filters.
- To familiarize fundamentals of shell (bash), shell programming, pipes, Control structures, arithmetic in shell interrupt processing, functions, debugging shell scripts.
- To impart fundamentals of file concepts kernel support for file, File structure related system calls (file API's).
- To facilitate students in understanding Inter process communication, semaphore and shared memory.
- To explore real-time problem solution skills in Shell programming.

Unit-I

Basic bash Shell Commands: Interacting with the shell-Traversing the file system-Listing files and directories-Managing files and directories-Viewing file contents. **Basic Script Building:**Using multiple commands-Creating a script file-Displaying messages-Using variables-Redirecting input and output-Pipes-Performing math-Exiting the script. **Using Structured Commands:**Working with the if-then statement-Nesting ifs-Understanding the test command-Testing compound conditions-Using double brackets and parentheses-Looking at case.

(Book-1, Chapters: 3, 11, and 12)

Unit-II

More Structured Commands: Looping with for statement-Iterating with the until statement-Using the while statement-Combining loops-Redirecting loop output. **Handling User Input:** Passing parameters-Tracking parameters-Being shifty-Working with options-Standardizing options-Getting user input. **Script Control:** Handling signals-Running scripts in the background-Forbidding hang-ups -Controlling a Job-Modifying script priority-Automating script execution.

(Book-1, Chapters: 13, 14, and 16)

Unit-III

Creating Functions: Basic script functions-Returning a value-Using variables in functions-Array and variable functions-Function recursion-Creating a library-Using functions on the command line. **Writing Scripts for Graphical Desktops:** Creating text menus-Building text window widgets-Adding X Window graphics. **Introducing sed and gawk:** Learning about the sed Editor-Getting introduced to the gawk Editor-Exploring sed Editor basics.

(Book-1, Chapters: 17, 18, and 19)

Unit-IV

Regular Expressions: Defining regular expressions-Looking at the basics-Extending our patterns-Creating expressions. **Advanced sed:** Using multiline commands-Understanding the hold space-Negating a command-Changing the flow-Replacing via a pattern-Using sed in scripts-Creating sed utilities. **Advanced gawk:** Reexamining gawk-Using variables in gawk-Using structured commands-Formatting the printing-Working with functions.

(Book-1, Chapters: 20, 21, and 22)

Unit-V

Working with Alternative Shells: Understanding the dash shell-Programming in the dash shell-Introducing the zsh shell-Writing scripts for zsh. **Writing Simple Script Utilities :** Automating backups-Managing user accounts-Watching disk space. **Producing Scripts for Database, Web, and E-Mail :** Writing database shell scripts-Using the Internet from your scripts-Emailing reports from scripts. **Using Python as a Bash Scripting Alternative:** Technical requirements-Python Language-Hello World the Python way-Pythonic arguments-Supplying arguments-Counting arguments-Significant whitespace-Reading user input-Using Python to write to files-String manipulation.

(Book-1, Chapters: 23, 24, 25, and Book-2, Chapter: 14)

Text book:

1. Richard Blum, Christine Bresnahan, "Linux Command Line and Shell Scripting BIBLE", Wiley Publishing, 3rd Edition, 2015. **Chapters:** 3, 11 to 14, 16 to 25.
2. Mokhtar Ebrahim, Andrew Mallett, "Mastering Linux Shell Scripting", Packt Publishing, 2nd Edition, 2018. **Chapter:** 14.

Reference Books:

1. CliffFlynt, SarathLakshman, ShantanuTushar, "Linux Shell Scripting Cookbook ", Packt Publishing, 3rd Edition, 2017.
2. Stephen G.Kochan, Patrick Wood, "Shell Programming in Unix, Linux, and OS X", Addison Wesley Professional, 4th Edition, 2016.
3. Robert Love, "Linux System Programming", O'Reilly Media, Inc, 2013
4. W.R. Stevens, "Advanced Programming in the UNIX environment", 2nd Edition, Pearson Education, 2013
5. Graham Glass, King Ables, " UNIX for Programmers and Users", 3rd Edition, Pearson Education, 2003

Course Outcomes

On the successful completion of the course, students will be able

CO1:	To understand, apply and analyze the concepts and methodology of Linux shell programming	K1-K6
CO2:	To comprehend, impart and apply fundamentals of control structure and script controls	K1-K6
CO3:	To understand, analyses and evaluate the functions, graphical desktop interface and editors	K1-K6
CO4:	To collaborate, apply and review the concepts and methodology of regular expression and advanced gawk	K1-K6
CO5:	To comprehend, use and illustrate the advance concepts such as alternate shell script, data connectivity and bash scripting using python	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C01	S	S	S	-	S	L	-	M	M	M	M	S
C02	S	S	M	-	S	L	-	M	M	M	M	S
C03	S	S	M	-	S	L	-	M	M	S	S	S
C04	S	S	M	-	S	L	-	M	M	M	M	S
C05	S	S	M	-	S	L	-	M	M	M	M	S

S- Strong; M-Medium; L-Low

Python Programming

Course Objectives:

- To acquire programming skills in core Python.
- To learn Strings and function.
- To develop object oriented skills in Python.
- To comprehend various Python Packages.
- To develop web applications using Django.

Unit I

Introduction : Fundamental ideas of Computer Science - Strings, Assignment, and Comments - Numeric Data types and Character sets - Expressions - Loops and Selection Statements: Definite iteration: the for Loop - selection: if and if-else statements - Conditional iteration: the while Loop.

Unit II

Strings and Text Files: Accessing Characters and substrings in strings - Data encryption-Strings and Number systems- String methods - Text - Lists and Dictionaries: Lists - Dictionaries - Design with Functions: A Quick review - Problem Solving with top-Down Design - Design with recursive Functions - Managing a Program's namespace - Higher-Order Functions.

Unit III

Design with Classes: Getting inside Objects and Classes - Data-Modeling Examples - Building a New Data Structure - The Two - Dimensional Grid - Structuring Classes with Inheritance and Polymorphism - GraphicalUser Interfaces - The Behavior of terminal-Based programs and GUI-Based programs - Coding Simple GUI-Based programs - Windows and Window Components - Command Buttons and responding to events.

Unit IV

Working with Python Packages: NumPy Library-Ndarray - Basic Operations - Indexing, Slicing and Iteration - Array manipulation - Pandas -The Series - The DataFrame - The Index Objects - Data Vizualization with Matplotlib - The Matplotlib Architecture - pyplot - The Plotting Window - Adding Elements to the Chart - Line Charts - Bar Charts - Pie charts.

Unit V

Django: Installing Django – Building an Application – Project Creation – Designing the Data Schema - Creating an administration site for models - Working with QuerySets and Managers – Retrieving Objects – Building List and Detail Views.

Text Book:

1. K.A. Lambert, "Fundamentals of Python: first programs", Second Edition, Cengage Learning, 2018 **(Unit - I, II and III)**
2. Fabio Nelli, "Python Data Analytics: With Pandas, NumPy, and Matplotlib", Second Edition, Kindle Edition, 2018 **(Unit - IV)**
3. Antonio Mele, "Django 3 By Example", Third Edition, 2020 **(Unit - V)**

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Comprehend the programming skills in python and develop applications using conditional branches and loop	K1- K6
CO2	Create python applications with strings and functions	
CO3	Understand and implement the Object Oriented Programming paradigm with the concept of objects and classes, Inheritance and polymorphism	
CO4	Evaluate the use of Python packages to perform numerical computations and data vizualization	
CO5	Design interactive web applications using Django	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	M	S	S	S	S	M	S	S
CO2	S	S	S	M	S	S	S	S	S	S	M	S
CO3	S	M	S	S	M	S	M	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S	M	S
CO5	S	S	S	S	S	S	S	S	S	M	M	S

S- Strong; M-Medium; L-Low

Data Engineering and Management

Course Objectives:

- To understand Data Management concepts.
- To get brief knowledge on Data Modeling.
- To analyse the techniques used in Distributed Databases.
- To assess Distributed database and Business Modelling.
- To get familiar with CRM tools.

Unit-I

DATABASE DEVELOPMENT: Database architecture of an information system- Overview of the database development process-Conceptual data modeling- Relational data analysis-Roles of a data model-Physical database design. DATA MANAGEMENT: Problems encountered without data management-Data management responsibilities-Data management activities-Roles within data management-Benefits of data management-Relationship between data management and enterprise.

Unit-II

CORPORATE DATA MODELLING: Need for a corporate data model-Nature of a corporate data model- Develop a corporate data model - Corporate data model principles. DATA DEFINITION AND NAMING: Elements of a data definition-Data naming conventions. DATA QUALITY: Issues associated with poor data quality-Causes of poor data quality-Dimensions of data quality-Data model quality-Improving data quality. DATA ACCESSIBILITY: Data security-Data integrity-Data recovery.

Unit-III:

USE OF PACKAGED APPLICATION SOFTWARE: Application software packages-Impact on data management. DISTRIBUTED DATA AND DATABASES: Rationale for distributing data-Perfect distributed database system-Top down fragmentation and partitioning. Bottom up integration-The management of replication. BUSINESS INTELLIGENCE: Data warehousing-Multidimensional model of data-Standard reporting tools-Online analytical processing OLAP-Relational schema for a data warehouse.

Unit-IV:

CRM: Three main pillars of CRM. GETTING TO KNOW YOUR CUSTOMER: 360-degree client view. UTILIZING ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING IN YOUR CRM STRATEGY: Evolution of AI-Current state of AI-Teaming up AI with people-Appling AI to your CRM solution-ethical aspects of AI-An example of AI in CRM processes.

Unit-V:

CLOUD VERSUS ON PREMISE VERSUS HYBRID: Factors influencing vendor selection-Hybrid deployment-what are your options. CRM DIFFERENTIATORS: It's not about the feature list; it's about the ecosystem-Fourth industrial revolution and CRM-AI and smart cloud-To cloud or not to cloud-Leveraging smart cloud into CRM-Big data-Social selling and advertising-Implementation tools-Sustainable CRM platform.

Text Books:

1. Keith Gordon, "Principles of Data Management Facilitating Information Sharing", BCS Learning, 2013. (Chapters:1-5, 7,8,12,13,14)
2. Max Fatouretchi, "The Art of CRM", Packt Publishing, 2019.(Chapters: 1,2,5,8,9)

Reference Books:

1. Peter Ghavami, "Big Data Management_ Data Governance Principles for Big Data Analytics", De Gruyter, 2020.
2. Francis Buttle, Stan Maklan, Customer Relationship Management Concepts and Technologies, Routledge, 2019.

Course Outcome:

On the successful completion of the course, students will be able to,

Course Outcome	Description	Knowledge Level
CO1	Comprehend the Data Management concepts and analyse the relationship with the enterprise	K1- K6
CO2	Analyze Data Modelling concepts and assess its quality	
CO3	Understand and implement business modelling techniques	
CO4	Evaluate the use of Artificial Intelligence and Machine Learning in CRM	
CO5	Develop CRM applications in cloud	

K1- Remember, K2 - Understand, K3 - Apply, K4 - Analyze,

K5 - Evaluate, K6 -Create

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	M	S	S	S	S	M	S	S
CO2	S	S	S	M	S	S	S	S	S	S	M	S
CO3	S	M	S	S	M	S	M	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S	M	S
CO5	S	S	S	S	S	S	S	S	S	M	M	S

L - Low, M- Medium, S – Strong

Data Engineering and Management Lab

Course Objectives:

- To acquire basic scripting knowledge in MongoDB.
 - To learn CRUD Operation on MongoDB database.
 - To comprehend MongoDB using DbVisualizer.
 - To be familiar with Zoho CRM features.
 - To customize your application using Zoho CRM.
1. Write a script to create a MongoDB database and perform insert operation.
 2. Write a MongoDB script to perform query operations.
 3. Write a MongoDB Script to perform update operations.
 4. Write a MongoDB Script to update documents with aggregation pipeline.
 5. Write a MongoDB script to delete single and multiple documents.
 6. Write a MongoDB script to perform string aggregation operations.
 7. Design a Data Model for MongoDB using DbVisualizer.
 8. Perform CRUD operations using DbVisualizer.
 9. Create a Zoho CRM account and organize your Tasks, Meetings and Deals.
 10. Create and maintain a project using Zoho CRM features.

Course Outcomes

On the successful completion of the course, students will be able to

Course Outcome	Description	Knowledge Level
CO1	Comprehend the scripting knowledge in MongoDB and perform basic operations in shell prompt	K1- K6
CO2	Implement, Create, Read, Update and Delete Operations on MongoDB database	
CO3	Analyze MongoDB using DbVisualizer	
CO4	Assess Zoho CRM features for managing the customer relationships	
CO5	Create a customized application in Zoho CRM	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	S	S	M	S	S	S	M	S	S
CO2	S	S	S	S	S	S	S	S	S	S	L	S
CO3	S	M	S	S	M	S	M	S	S	S	S	S
CO4	S	S	S	M	S	S	S	L	S	S	M	S
CO5	S	S	S	S	M	S	S	S	S	S	S	S

S- Strong; M-Medium; L-Low

Architecture and Frameworks

Course Objectives

- To understand the basics, benefits and purpose of software architecture.
- Understand the quality attributes to fulfill the software requirements and relates the software with an organization.
- Explore the design patterns, best practice and paradigms of efficient software development.
- Understand the performance and security measures of software architecture.
- Enable the developers to advance their carrier in software domain.

Unit – I

Software architecture introduction – Importance of Software architecture – Software architecture consumers – Architect role - software architecture in an organization – Types of software architects – Software development methodologies – Project management – Office politics – Software risk management – Configuration management – Software product lines.

Unit – II

Domain Knowledge – Developing business acumen – Domain-driven design – requirement engineering – requirement elicitation –Software Quality attributes: Maintainability – Usability –Availability – Portability – Interoperability – Testability.

Unit – III

Software Architectures design – Importance - Top-down Versus bottom-up design approaches – Architectural drivers – Documenting the Software architecture design – Systematic approach - Attribute-driven design – Microsoft's technique for architecture and design –Architecture-centric design method – Architecture development method – Tracking the progress of the software architecture's design.

Unit – IV

Designing orthogonal software systems – Minimizing complexity – SOLID design principles – Software architecture patterns – layered – Event-driven architecture – Model-View patterns – Service-oriented architecture.

Unit – V

Architecting Modern Applications.- Importance of Performance – Performance improvement - Server side caching – Web application performance – Database performance -Securing software systems – Threat modelling – Secure by design.

Text Book

1. Joseph Ingeno, "Software Architect's Handbook" Packt Publishing 2018.

Reference books

1. Oliver Vogel, Indo Arnold, ArifChughtaiandTimoKehrer, "Software Architecture" Springer-Verlag, 2011.
2. Ian Gorton, "Essential Software architecture", Second Edition, Springer, 2011.
3. Len Bass, Paul Clements and Rick Kazman, "Software architecture in practice", Third edition, Addison-Wesley, 2013.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Understand, analyze and evaluate the purpose of Software architecture and development methodologies with consideration of risk management	K1-K6
CO2	Comprehend, apply and evaluate the domain knowledge for software development process and determine the impact of quality attributes.	K1-K6
CO3	Understand, track and examine the systematic approach for various software design models with effective document process	K1-K6
CO4	Illustrate and summarize the functions of orthogonal systems with complexity, design principles and design pattern for software architecture	K1-K6
CO5	Comprehend, analyze and evaluate the performance and security measures for Server, Web and Database applications in order to create the secure software systems for various domain applications	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- evaluate and K6- Create

Mapping with Programme Outcome

	PO1	PO 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	PO 10	PO 11	PO 12
CO1	M	S	-	L	M	S	M	M	L	S	-	-
CO2	M	S	-	M	M	L	S	S	M	M	S	-
CO3	S	M	-	S	S	M	S	M	M	S	M	-
CO4	S	M	L	S	M	L	S	L	S	M	S	M
CO5	M	S	M	L	S	L	M	S	L	S	-	M

S- Strong; M-Medium; L-Low

Architecture and Frameworks - Lab

Course Objectives

- To understand and implement the basic concepts of Software architecture and its functions.
- To acquire programming skills to develop Implement various technologies and services associated with network protocols along with the challenges of data transfer.
- Implement the importance and functioning of Routing Protocols over communication service.
- To acquire skills to connect two routers and any two switches.
- To comprehend related to SSH protocols and accessing the remote device.

Note: Use the solid servers and client specification for implementation

<https://github.com/solid/specification/>.

Implement the following using Linux / Windows environments

1. Find the WebID profile document and display the necessary attributes.
2. Set and access the primary authentications with account recovery mechanisms.
3. Set and access the secondary authentications with account recovery mechanisms.
4. Design authorization and web access control.
5. Find the content representation.
6. Reading resources from HTTP REST API and WebSockets API.
7. Writing resources from HTTP REST API and WebSockets API.
8. Data notification using Social Web App protocol.
9. Managing subscriptions and friends list using Social Web App protocol.
10. Managing list of followers and following list using Social Web App Protocol.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Comprehend the programming skills of Software architecture tools and packages	K1-K6
CO2	Understand and implement the user profiles and authentication with recovery mechanism.	K1-K6
CO3	Comprehend and evaluate the access control and content representation use of FTP server	K1-K6
CO4	Understand and implement reading and writing resources for various applications	K1-K6
CO5	Identify and examine the notifications, friends, and follower list of social application protocols.	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping Course outcomes with Programme outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO9	PO10	PO11	PO12
CO 1	S	S	M	L	M	S	-	-	-	-	-	-
CO 2	S	M	S	S	S	M	-	-	-	-	-	-
CO 3	S	M	S	M	S	M	-	-	-	-	-	-
CO 4	S	M	L	S	M	L	-	-	-	-	-	-
CO 5	M	S	M	L	S	L	-	-	-	-	-	-

S- Strong; M-Medium; L-Low

Software Development Technologies

Course Objectives

- To learn and Implementing Micro services.
- To analysing the Azure Kubernetes Service.
- To learn and analyse .NET DevOps for Azure and its applications.
- To building code for .NET core applications.
- To get familiarized with Azure pipelines.

Unit -I

Implementing Microservices: Client to microservices communication, Interservice communication, data considerations, security, monitoring, microservices hosting platform options. Azure Service Fabric: Introduction, core concepts, supported programming models, service fabric clusters, develop and deploy applications of service fabric. Monitoring Azure Service Fabric Clusters: Azure application, resource manager template, Adding Application Monitoring to a Stateless Service Using Application Insights, Cluster monitoring, Infrastructure monitoring.

Unit-II

Azure Kubernetes Service (AKS): Introduction to kubernetes and AKS, AKS development tools, Deploy applications on AKS. Monitoring AKS: Monitoring, Azure monitor and analytics, monitoring AKS clusters, native kubernetes dashboard, Prometheus and Grafana. Securing Microservices: Authentication in microservices, Implenting security using API gateway pattern, Creating application using Ocrlot and securing APIs with Azure AD. Database Design for Microservices: Data stores, monolithic approach, Microservices approach, harnessing cloud computing, database options on MS Azure, overcoming application development challenges. Building Microservices on Azure Stack: Azure stack, Offering IaaS, PaaS on-premises simplified, SaaS on Azure stack.

Unit-III

.NET DevOps for Azure: DevOps introduction, Problem and solution. Professional Grade DevOps Environment: The state of DevOps, professional grade DevOps vision, DevOps architecture, tools for professional DevOps environment, DevOps centered application. Tracking work: Process template, Types of work items, Customizing the process, Working with the process. Tracking code: Number of repositories, Git repository, structure, branching pattern, Azure repos configuration, Git and Azure.

Unit-IV

Building the code: Structure of build, using builds with .NET core and Azure pipelines, Validating the code: Strategy for defect detection, Implementing defect detection. Release candidate creation: Designing release candidate architecture, Azure artifacts workflow for release candidates, Deploying the release: Designing deployment pipeline, Implementing deployment in Azure pipelines. Operating and monitoring release: Principles, Architectures for observability, Jumpstarting observability.

Unit-V

Introduction to APIs: Introduction, API economy, APIs in public sector. API Strategy and Architecture: API Strategy, API value chain, API architecture, API management. API Development: Considerations, Standards, kick-start API development, team orientation. API Gateways: API Gateways in public cloud, Azure API management, AWS API gateway. API Security: Request-based security, Authentication and authorization.

Text Book

1. Harsh Chawla and Hemant Kathuria, Building Microservices Applications on Microsoft Azure- Designing, Developing, Deploying, and Monitoring, Apress, 2019.
2. Jeffrey Palermo , NET DevOps for Azure A Developer's Guide to DevOps Architecture the Right Way, Apress, 2019.
3. Thurupathan and Vijayakumar, Practical API Architecture and Development with Azure and AWS - Design and Implementation of APIs for the Cloud, Apress, 2018.

Reference Books

1. Karl Matthias and Sean P. Kane, Docker: Up and Running, O'Reilly Publication, Second Edition 2018.
2. Len Bass, Ingo Weber, Liming Zhu, "DevOps, A Software Architects Perspective", Addison Wesley-Pearson Publication, First Edition 2015.
3. John Ferguson Smart, "Jenkins, The Definitive Guide", O'Reilly Publication, First Edition 2011.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	To understand, apply and summarize the basic concepts of Micro services communication Microsoft Azure and Dev Ops for software development life cycle	K1-K6
CO2:	To illustrate, and implement Azure Kubernetes Service tools for software development life cycle	K1-K6
CO3:	To recognize, analyse and summarize the functionalities of .NET Dev Ops for Azure applications	K1-K6
CO4:	To understand, design and evaluate the principles and architecture service tools for software development life cycle.	K1-K6
CO5:	To comprehend, implement and review the functionalities of API and API gateways for cloud and Azure applications	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	-	M	-	-	-	S	-	-	-
CO2	S	S	M	-	M	-	-	-	S	-	-	-
CO3	S	S	S	-	S	-	-	-	S	S	S	S
CO4	S	S	M	-	M	-	-	-	S	-	-	-
CO5	S	S	M	-	M	-	-	-	S	-	-	-

S- Strong; M-Medium; L-Low

Software Development Technologies Lab

Course Objectives

- To understand the concept of DevOps with associated technologies and methodologies.
- To be familiarized with Jenkins, which is used to build & test software Applications.
- To understand Continuous integration in Devops environment.
- To understand Docker to build, ship and run containerized images.
- To use Docker to deploy and manage Software applications running on Container.

List of Programs

1. Deploy Version Control System / Source Code Management, install git and create a GitHub account.
2. Perform various GIT operations on local and Remote repositories using GIT Cheat-Sheet.
3. Continuous Integration: install and configure Jenkins with Maven/Ant/Gradle to setup a build Job.
4. Build the pipeline of jobs using Maven / Gradle / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.
5. Implement Jenkins Master-Slave Architecture and scale your Jenkins standalone implementation by implementing slave nodes.
6. Setup and Run Selenium Tests in Jenkins Using Maven.
7. Implement Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.
8. Implement Dockerfile instructions, build an image for a sample web application using Dockerfile.
9. Install and Configure Pull based Software Configuration Management and provisioning tools using Puppet.
10. Implement LAMP/MEAN Stack using Puppet Manifest.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	To Understand and analyse the importance of Jenkins to Build, Deploy and Test Software Applications	K1-K6
CO2:	To synthesis and summarize the importance of Software Configuration Management in DevOps	K1-K6
CO3:	To identify, analyze and illustrate the Containerization of OS images and deployment of applications over Docker	K1-K6
CO4	To design, analyze and develop the Pull based Software Configuration Management	K1-K6
CO5	To design, analyze and develop Puppet Manifest	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	-	M	-	-	-	S	S	S	S
CO2	S	S	M	-	M	-	-	-	S	S	S	S
CO3	S	S	M	-	M	-	-	-	S	S	S	S
CO2	S	S	M	-	M	-	-	-	S	S	S	S
CO3	S	S	M	-	M	-	-	-	S	S	S	S

S- Strong; M-Medium; L-Low

Soft Computing

COURSE OBJECTIVES:

- Develop the skills to gain a basic understanding of neural network theory and fuzzy logic theory.
- To understand supervised and unsupervised learning algorithms.
- To enable the students to gain a basic understanding of neural networks.
- To know about fuzzy logic, fuzzy inference systems, and their functions.
- To impart basic knowledge on Genetic algorithms and their applications.

UNIT-I

INTRODUCTION TO SOFT COMPUTING: Artificial Neural Networks- Biological Neurons- Basic Models of Artificial Neural Networks-Connections- Learning-Activation Functions- Important Terminologies of ANNs- Muculloch and Pitts Neuron-Linear Separability- Hebb Network-Flowchart of Training Process-Training Algorithm.

UNIT - II

SUPERVISED LEARNING NETWORK : Perceptron Networks–Perceptron Learning Rule-Architecture-Flowchart for Training Process-Perceptron Training Algorithms for Single Output Classes-Perceptron Training Algorithm for Multiple Output Classes-Perceptron Network Testing Algorithm - Adaptive Linear Neuron-Delta Rule for Single Output Unit-Flowchart for training algorithm-Training Algorithm – Testing Algorithm - Multiple Adaptive Linear Neurons-Architecture-Flowchart of Training Process-Training Algorithm-Back Propagation Network-Architecture-Flowchart for Training Process-Training Algorithm-Learning Factors of Back-Propagation Network-Radial Basis Function Network- Architecture-Flowchart for Training Process-Training Algorithm.

UNIT-III

UNSUPERVISED LEARNING NETWORK: Associative Memory Networks - Auto Associative Memory Network-Architecture-Flowchart for Training Process-Training Algorithm-Testing Algorithm- Bidirectional Associative Memory- Architecture-Discrete Bidirectional Associative Memory-Iterative Auto Associative Memory Networks - Linear AutoAssociative Memory-Kohonen Self-Organizing Feature Map- Architecture-Flowchart for Training Process-Training Algorithm.

UNIT-IV

INTRODUCTION TO FUZZY LOGIC: Classical Sets –Operations on Classical Sets-Fuzzy sets - Fuzzy Sets- Properties of Fuzzy Sets- Fuzzy Relations – Membership Functions: Fuzzification- Methods of Membership Value Assignments – Defuzzification – Lambda-Cuts for Fuzzy sets and Fuzzy Relations – Defuzzification Methods–Max-Membership Principle-Centroid Method-Weighted Average Method-Mean Max Membership-Center of Sums-Center of Largest Area-First of Maxima - Fuzzy Set Theory - Fuzzy Arithmetic And Fuzzy Measures: Fuzzy Measures – Belief and Plausibility Measures-Probability Measures-Possibility and Necessity Measures- Formation of Rules –Fuzzy Inference Systems (FIS) – Fuzzy Decision Making – Fuzzy Logic Control Systems.

UNIT-V

GENETIC ALGORITHM: Introduction - Biological Background - Traditional Optimization and Search Techniques -Gradient Based Local Optimization Method-Random Search-Stochastic Hill Climbing-Simulated Annealing-Symbolic Artificial Intelligence-Operators in Genetic Algorithm -Encoding-Selection-Crossover-Mutation - Stopping Conditions for Genetic Algorithm Flow-Genetic Programming-Working of Genetic Programming-Characteristics of Genetic Programming-Data Representation.

TEXT BOOKS

1. Principles of Soft Computing, S.N. Sivanandam, S.N.Deepa, Wiley, Third Edition, 2019.

UNIT I: Chapter 1: 2.1,2.3,2.4,2.5,2.6,2.7

UNIT II: Chapter 2: 3.2,3.3,3.4,3.5,3.6

UNIT III: Chapter 3: 4.3,4.4,4.7,5.3

UNITIV:Chapter4:7.2,7.3,8.4,9.3,9.4,10,10.2,10.3,10.4,11.4,12.8,14

UNIT V: Chapter 5: 15,15.2,15.3,15.4,15.9,15.10

REFERENCE BOOKS

1. Das, A. (2018). Artificial Intelligence and Soft Computing for Beginners.
2. Amit, K. (2018). Artificial intelligence and soft computing: behavioral and cognitive modeling of the human brain. CRC press.
3. Rajasekaran, S., &Pai, G. V. (2011). Neural networks, fuzzy logic and genetic algorithm: synthesis and applications (with cd). PHI Learning Pvt. Ltd.
4. Jang, J. S. R., Sun, C. T., &Mizutani, E. (2004). Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Transactions on automatic control, 42(10), 1482-1484.
5. Gupta, M. M. (2004). Soft computing and intelligent systems: theory and applications. Elsevier.
6. Jang, J. S. R., Sun, C. T., &Mizutani, E. (1997). Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Transactions on automatic control, 42(10), 1482-1484.

COURSE OUTCOMES:

- On the successful completion of the course, students will be able to

CO1	To provide an introduction to the basic principles, techniques, and applications of soft computing	K- 1 K2	LO
CO2	To get familiar with Neural network architectures and supervised learning algorithms	K3	IO
CO3	To understand the architectures and algorithms of Unsupervised Learning techniques	K3- K4	H O
CO4	Develop the skills to gain a basic understanding of fuzzy logic theory and fuzzy inference systems	K4	IO
CO5	Ability to learn traditional optimization and search techniques and genetic programming	K5	H O

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	M	S	M	S	S	S	M	S	M	S	S
CO2	M	S	M	S	M	M	M	S	M	S	S	M
CO3	M	M	S	S	S	M	M	S	S	S	S	S
CO4	S	S	M	M	M	S	S	S	S	M	M	M
CO5	S	S	S	S	S	M	S	M	M	S	S	M

S- STRONG; M-MEDIUM; L-LOW

Soft Computing Lab

COURSE OBJECTIVES:

- To implement various Supervised Neural Network-based approaches.
- To apply the fuzzy-based logical operations and arithmetic operations.
- To implement unsupervised neural network approaches.
- To solve a problem using a simple genetic algorithm.
- To implement logic gates.

Program List

1. Implementation of Logic gates using Artificial Neural Network.
2. Implementation of Perception Algorithm.
3. Implementation of Back Propagation Algorithm.
4. Implementation of Self Organizing Maps.
5. Implementation of Radial Basis Function Network.
6. Implementation of De-Morgan's Law.
7. Implementation of McCulloch Pits Artificial Neuron model.
8. Implementation of Simple genetic algorithm.
9. Implementation of fuzzy based Logical operations.
10. Implementation of fuzzy based arithmetic operations.

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1	To apply supervised learning algorithms for real datasets	K- 1K2	LO
CO2	To implement Unsupervised Learning techniques	K3	IO
CO3	To apply fuzzy based arithmetic and logical operations	K3- K4	HO
CO4	To find solutions for problems using Genetic algorithm	K4	IO
CO5	To implement DeMorgan's Law	K5	HO

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	M	S	M	S	S	S	M	S	M	S	S
CO2	M	S	M	M	M	M	M	S	M	S	M	M
CO3	M	M	S	S	S	M	M	S	S	S	S	S
CO4	S	S	S	M	M	S	S	M	S	S	S	M
CO5	S	S	S	S	S	M	M	M	M	S	M	M

S- STRONG; M-MEDIUM; L-LOW

Linux and Shell Programming - Lab

Course Objectives

- To enable the students to study and understand the efficiency of Linux shell script.
- To demonstrate the File Backup process.
- To develop and implement the shell script for GUI processing.
- To develop and implement the shell script for IPC and Networking.
- To demonstrate PostgreSQL.

List of Programs

1. Write a Shell Script program to calculate the number of days between two dates.
2. Write a Shell Script program to check systems on local network using control structures with user input.
3. Write a Shell Script program to check systems on local network using control structures with file input.
4. Write a Shell Script program to demonstrate the script control commands.
5. Write a Shell Script program to demonstrate the Shell script function.
6. Write a Shell Script program to demonstrate the Regular Expressions.
7. Write a Shell Script program to demonstrate the sed and awk Commands.
8. Write a Shell Script program to demonstrate the File Backup process through creating a daily archive location.
9. Write a Shell Script program to create a following GUI tools.
 - a) Creating text menus
 - b) Building text window widgets
10. Write a Shell Script program to demonstrate to connect a PostgreSQL Data base and performing CRUD operations.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	To understand, apply and analyze the concepts and methodology of Linux shell programming	K1-K6
CO2:	To comprehend, impart and apply fundamentals of control structure and script controls	K1-K6
CO3:	To understand, analyses and evaluate the functions, graphical desktop interface and editors	K1-K6
CO4:	To collaborate, apply and review the concepts and methodology of regular expression and advanced gawk	K1-K6
CO5:	To comprehend, use and analyze the advance concepts such as alternate shell script, dy and bash scripting using PostgreSQL	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	-	-	-	M	-	-	-
CO2	S	S	S	-	S	-	-	-	M	-	-	-
CO3	S	S	S	-	S	-	-	-	M	S	S	S
CO4	S	S	S	-	S	-	-	-	M	-	-	-
CO5	S	S	S	-	S	-	-	-	M	S	S	S

S- Strong; M-Medium; L-Low

Python Programming Lab

Course Objectives:

This course enables the students:

- To master the fundamentals of writing python scripts.
- To create program using elementary data items.
- To implement Python programs with conditionals and loops.
- To use functions for structuring Python programs.
- To develop web programming with Django.

Implement the following in Python:

1. Program using elementary data items, lists, dictionaries and tuples.
2. Program using conditional branches, loops.
3. Program using functions.
4. Program using classes and objects.
5. Program using inheritance.
6. Program using polymorphism.
7. Program using Numpy.
8. Program using Pandas.
9. Program using Matplotlib.
10. Program for creating dynamic and interactive web pages using forms.

Course Outcomes

On the successful completion of the course, students will be able to

C01	Comprehend the programming skills in python and write scripts	K1- K6
C02	Create python applications with elementary data items, lists, dictionaries and tuples	
C03	Implement the Object Oriented Programming programming concepts such as objects and classes, Inheritance and polymorphism	
C04	Assess the use of Python packages to perform numerical computations and perform data vizualization	
C05	Create interactive web applications using Django	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C01	S	S	M	S	S	S	S	S	S	M	S	S
C02	S	S	S	S	S	S	S	S	S	S	M	S
C03	S	S	S	S	S	S	M	S	S	M	L	S
C04	S	S	S	S	S	S	S	M	S	S	S	S
C05	S	S	S	S	L	S	M	S	S	M	M	S

S- Strong; M-Medium; L-Low