ST.JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

Choondacherry P.O., Pala, Kottayam - 686579
Kerala, India

SYLLABUS

MCA(INTEGRATED)

SEMESTER 5- 10

-2024 SCHEME-

www.sjcetpalai.ac.in

Signed By: Devassia V P
Reason: Approved by SJCET Academic Council held on 25.11.2025

Programme Outcomes (POs)

PO1(Foundation Knowledge): Apply knowledge of mathematics, programming logic, and coding fundamentals for solution architecture and problem-solving.

PO2 (Problem Analysis): Identify, review, formulate, and analyze problems primarily focusing on customer requirements using critical thinking frameworks.

PO3 (Development of Solutions): Design, develop, and investigate problems with an innovative approach for solutions incorporating ESG/SDG goals.

PO4 (Modern Tool Usage): Select, adapt, and apply modern computational tools, such as the development of algorithms, with an understanding of the limitations, including human biases.

PO5 (Individual and Teamwork): Function and communicate effectively as an individual or team leader in diverse and multidisciplinary groups, using methodologies such as agile.

PO6 (Project Management and Finance): Apply the principles of project management, including scheduling and work breakdown structure, and be knowledgeable about finance principles for profitable project management.

PO7 (Ethics): Commit to professional ethics in managing software projects, especially in financial aspects. Learn to use new technologies for cybersecurity and insulate customers from malware.

PO8 (Life-long Learning): Continuously enhance management skills and the ability to learn, keeping up with contemporary technologies and ways of working.

INDEX

CURRICULUM

INT MCA SEMESTER V									
Course No	Course	Hours/week			IA Marks	ESE Marks	Total	Credits	Exam Slot
		L	T	P					
24SJINMCA301	Numerical Methods	3	1	-	40	60	100	4	A
24SJINMCA303	User Interface Design	3	1	-	40	60	100	4	B
24SJINMCA305	Introduction to RDBMS and SQL	3	1	-	40	60	100	4	C
24SJINMCA307	Fundamentals of Information Systems Security	3	1	-	40	60	100	4	D
24SJINMCA309	Introduction to Operations Research	3	1	-	40	60	100	4	E
24SJINMCA331	RDBMS Lab			4	100		100	1	S
24SJINMCA333	User Interface Design Lab			4	100		100	1	T
		15	**5**	**8**	**400**	**300**	**700**	**22**	

INT MCA SEMESTER VI									
Course No	Course	Hours/week			IA Marks	ESE Marks	Total	Credits	Exam Slot
		L	T	P					
24SJINMCA302	Introduction to Software Engineering	3	1	-	40	60	100	4	A
24SJINMCA304	Fundamentals of Data Science	3	1	-	40	60	100	4	B
24SJINMCA306	Visual Programming	3	1	-	40	60	100	4	C
24SJINMCA308	Design & Analysis of Algorithms	3	1	-	40	60	100	4	D
24SJINMCA310	Mini Project 1/Internship			4	100		100	2	R
24SJINMCA332	Open Source Platforms - Lab		1	3	100		100	1	S
24SJINMCA334	Visual Programming Lab			4	100		100	1	T
		12	**5**	**11**	**460**	**240**	**700**	**20**	

INT MCA SEMESTER VII

Course No	Course	Hours/week			IA Marks	ESE Marks	Total	Credits	Exam Slot
		L	T	P					
24SJINMCA401	M- Commerce	3	1	-	40	60	100	4	A
24SJINMCA403	Java Programming	3	1	-	40	60	100	4	B
24SJINMCA405	Advanced Web programming	3	1	-	40	60	100	4	C
24SJINMCA407	Advanced Software Engineering	3	1	-	40	60	100	4	D
24SJINMCA409	Distributed Computing	3	1	-	40	60	100	4	E
24SJINMCA431	Java Programming - Lab		1	3	100		100	1	S
24SJINMCA433	Advanced Web programming Lab		1	3	100		100	1	T
24SJINMCANC1	Entrepreneurship & Innovations in Technology			1				0	
		15	**7**	**7**	**400**	**300**	**700**	**22**	

INT MCA SEMESTER VIII

Course No	Course	Hours/week			IA Marks	ESE Marks	Total	Credits	Exam Slot
		L	T	P					
24SJINMCA402	Advanced Database Management Systems	3	1	-	40	60	100	4	A
24SJINMCA404	Advanced Computer Networks	3	1	-	40	60	100	4	B
24SJINMCA406	Research Methodology	3	1	-	40	60	100	4	C
24SJINMCA4--	Elective 1	3	1	-	40	60	100	4	D
24SJINMCA4--	Elective 2	3	1	-	40	60	100	4	E
24SJINMCA432	Advanced DBMS Lab		1	3	100		100	1	S
24SJINMCA434	Networking & System Administration Lab		1	3	100		100	1	T
24SJINMCANC2	Industrial Readiness Training			1				0	
		15	**7**	**7**	**400**	**300**	**700**	**22**	

INT MCA SEMESTER IX									
Course No	Course	Hours/week			IA Marks	ESE Marks	Total	Credits	Exam Slot
		L	T	P					
24SJINMCA501	Data Science & Machine Learning	3	1	-	40	60	100	4	A
24SJINMCA503	Mobile Computing	3	1	-	40	60	100	4	B
24SJINMCA5--	Elective 3	3	1	-	40	60	100	4	C
24SJINMCA5--	Elective 4	3	1	-	40	60	100	4	D
24SJINMCA509	Mini Project 2			4	100		100	2	R
24SJINMCA531	Data Science Lab		1	3	100		100	1	S
24SJINMCA533	Mobile Application Development Lab		1	3	100		100	1	T
24SJINMCANC3	Domain Expertise Workshops			1				0	
		12	**6**	**11**	**460**	**240**	**700**	**24**	

INT MCA SEMESTER X									
Course No	Course	Hours/week			IA Marks	ESE Marks	Total	Credits	Exam Slot
		L	T	P					
24SJINMCA502	Comprehensive Viva	-	-	-	-	100	100	6	
24SJINMCA504	Seminar	-	-	2	50	-	50	2	
24SJINMCA506	Main Project/Internship	-	-	27	70	3 0	100	12	
		-	**-**	**29**	**120**	**130**	**250**	**20**	
	TOTAL						**6550**	**214**	

ST.JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

MCA Integrated

SEMESTER 5

24SJINMCA301	NUMERICAL METHODS	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course provides an introduction to the basic concepts and techniques of matrices and its operations, Numerical solution of Linear and Non-Linear equations, Curve Fitting, Numerical integration and differentiation.
Prerequisite:	NIL

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Identify special type of matrices and Elementary row operations of a matrix.	K5
CO2	Apply the concept of rank of the matrix to solve system of linear equations.	K5
CO3	Find Eigen values and Eigen vectors of a matrix numerically.	K5
CO4	Apply concept of curve fitting to fit the best curve using experimental observations.	K4
CO5	Describe interpolation and approximations.	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	2	1			
CO2	3	3	2	2				
CO3	3	3	3	3				
CO4	2	3	3	3	1			1
CO5	2	3	2	3				2

Mark distribution:

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Identify whether the matrix A= $\begin{matrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 3 \end{matrix}$ is symmetric?
2. Show that AB ≠ BA for the matrices A= [1 2 3] and B = $\begin{matrix} 1 & 2 \\ 5 & 3 \\ 2 & -1 \end{matrix}$
3. Illustrate the row Echelon form of the matrix A = $\begin{matrix} 2 & 4 & -2 \\ 4 & 9 & -3 \\ -2 & -1 & 7 \end{matrix}$
4. Evaluate A^{-1} where A = $\begin{matrix} -3 & -1 & -1 \\ -1 & -3 & 1 \\ -1 & 1 & -3 \end{matrix}$

Course Outcome 2 (CO2):

1. Identify whether the vectors (1,2,1), (4,1,2), (6,5,4), (-3,8,1) are linearly dependent?
2. Solve the system of linear equations 2x + 3y + 4z = 11, x + 5 y + 7z = 15, 3x +11y +13z = 25, 2x +y +z = 5.
3. Determine the rank of the matrix A = $\begin{matrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 1 & 1 & 1 \end{matrix}$.
4. Evaluate the values of x, y and z of the equations 45x + 2 y + 3z = 58, – 3x+22y+2z=47, 5x +y + 20z = 67 using Gauss Seidel method.

Course Outcome 3 (CO3):

1. Determine the Eigen values of A = $\begin{matrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{matrix}$, given that 1 is an Eigen value of A?
2. Determine the Eigen values and Eigen vectors of the matrix A = $\begin{matrix} 6 & -2 & 2 \\ -2 & 3 & -1 \\ 2 & -1 & 3 \end{matrix}$
3. Examine whether the matrix A = $\begin{matrix} 2 & 1 & -1 \\ 1 & 1 & -2 \\ -1 & -2 & 1 \end{matrix}$ is diagonalizable?
4. Evaluate the largest Eigen value and the corresponding Eigen vector of the matrix A = $\begin{matrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 2 \end{matrix}$

Course Outcome 4 (CO4):

1. Define principle of least squares.
2. Explain the normal equations of y = ax^2+ $\frac{b}{x}$.

3. Apply principle of least squares to fit a straight line to the data given below:

X	1	2	3	4	5
Y	14	27	40	55	68

4. Analyse the following data to fit a second degree polynomial of the form y = a + bx + cx^2 :

X	1	2	3	4	5	6	7	8	9
Y	2	6	7	8	10	11	11	10	9

Course Outcome 5 (CO5):

1. Identify the polynomial using Lagrange's Interpolation formula, given: y (1) = -3, y (3) = 9, y (4) = 30, y (6) = 132.
2. Calculate f(1.2) using Newton's Forward Interpolation Formula from the following data:

x	1	1.4	1.8	2.2
f(x)	3.49	4.82	5.96	6.5

3. Analyse the data given below to find sin 450 using Gauss Backward Formula:

θ	20	30	40	50	60	70	80
$sin\theta$	0.34202	0.502	0.64279	0.76604	0.86603	0.93969	0.98481

4. Evaluate U(3) given that U(1) = -26, U(2) = 12, U(4) = 256, U(6) = 844 using Newton's Divided Difference Formula.

Syllabus

Module 1 (Matrices)

Basic concepts, Matrix multiplication, Special matrices, Elementary row operations, Inverse of a matrix using Gauss Jordan Method

Module 2 (Linear Independence)

Linear dependence and Independence of vectors, Rank of a matrix, Homogeneous and nonhomogeneous linear equations-uniqueness and consistency-Gauss Elimination Method, Gauss Seidel Method.

Module 3 (Eigen values and Eigen vectors)

Eigen values and Eigen vectors, Properties of Eigen values and Eigen vectors, Diagonalization, Eigen values and Eigenvectors of a matrix by power method.

Module 4 (Curve Fitting)

Introduction, Principle of Least Squares, Method of Least Squares, Fitting of Linear Equations, Fitting second degree polynomial function.

Module 5 (Interpolation and Approximation)

Interpolation with equal intervals - Newton's forward and backward Interpolation, Gauss's forward and backward interpolation, Interpolation with unequal Intervals-Lagrange's Interpolation formula and Newton's Divided difference Formula.

Textbooks

1. Richard Bronson and Gabriel B. Costa, *Linear Algebra: An Introduction*, 2nd Edition, Elsevier Publishers, 2007.
2. S. R. K. Iyengar and R. K. Jain, *Numerical Methods*, New Age International Publishers, 2014.
3. S. C. Gupta and V. K. Kapoor, *Fundamentals of Mathematical Statistics*, 10th Revised Edition, S. Chand & Sons, Educational Publishers, New Delhi, 2000.

Reference Text Books

1. T. Veerarajan and T. Ramachandran, *Numerical methods with programming in 'C'*, Second Edition, Tata McGraw-Hill Publishing. Co. Ltd, 2007.
2. C. F. Gerald and P.O. Wheatley, *Applied Numerical Analysis*, 6th Edition, Pearson Education Asia, NewDelhi,2006.
3. M.K Jain, R.K Iyengar, R.K Jain, *Numerical Methods for Scientific and Engineering Computation*, New Age International Hill Publishers, NewDelhi-1997.

Web Resources

1. https://www.coursera.org/learn/numerical-methods-engineers
2. https://onlinecourses.nptel.ac.in/noc21_ma45/preview
3. https://www.khanacademy.org/math/statistics-probability/describing-relationships-quantitatively
4. https://www.khanacademy.org/math/linear-algebra

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Module 1**	**9 hours**
1.1	Definition, Operations on matrix – Matrix Addition, subtraction,	2
1.2	Matrix multiplication	1
1.3	Special Matrices	2

1.4	Elementary row operations, Row Echelon form	2
1.5	Inverse of a matrix using Gauss Jordan Method	2
2	**Module 2**	**10 hours**
2.1	Linear dependence and Independence of vectors	2
2.2	Rank of a matrix	2
2.3	Homogeneous and non-homogeneous linear equations	1
2.4	Uniqueness and consistency	1
2.5	Gauss Elimination method	2
2.6	Iterative methods of Solution: Gauss-Seidel method	2
3	**Module 3**	**10 hours**
3.1	Eigen values and Eigen vectors	3
3.2	Properties of Eigen values and Eigen vectors	2
3.3	Diagonalization	3
3.4	Eigen values and Eigenvectors of a matrix by power method.	2
4	**Module 4**	**9 hours**
4.1	Introduction	2
4.2	Principle of Least Squares	2
4.3	Method of Least Squares	2
4.4	Fitting of Linear Equations	2
4.5	Fitting second degree polynomial function	1
5	**Module 5**	**10 hours**
5.1	Introduction	1
5.2	Newton's forward and backward Interpolation	2
5.3	Gauss's forward and backward interpolation	2
5.4	Lagrange's Interpolation formula	2
5.5	Newton's Divided difference Formula	3

24SJINMCA303	USER INTERFACE DESIGN	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course introduces students to develop skills to create server-side scripts using PHP. Introduces server-side programming concepts and terminology. Explores a variety of server-side techniques and MySQL database manipulation.
Prerequisite:	Nil

Course Outcomes: After the completion of the course the student will be able to:		K Level
CO1	Identify and define key terms related to user interfaces and user interface design implementation	K2
CO2	Identify and describe common abstract user interface components, such as menu selection and dialog boxes	K2
CO3	Distinguish PHP as a server-side programming language	K5
CO4	Know about the basic concepts of Functions, Array and Strings in PHP	K5
CO5	Outline the principles behind using MySQL as a backend DBMS with PHP	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	1		2				2
CO2	3	2		2				2
CO3	3	2	2	3				2
CO4	3	2	2	3				2
CO5	3	2	3	3	2	2		2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain the Four Pillars of Design
2. List the steps involved in the contextual design method?

Course Outcome 2 (CO2):

1. Explain Direct manipulation
2. Explain the elements of form fill-in design

Course Outcome 3(CO3):

1. Is PHP case sensitive? Justify your answer
2. Demonstrate with an example explain data types in PHP

Course Outcome 4 (CO4):

1. Convert strings and arrays using implode and explode
2. Compare include and require statement in PHP

Course Outcome 5 (CO5):

1. Create a procedure to perform database connectivity in PHP?
2. Illustrate with an example to explain how data is stored in Sessions.

Syllabus

Module 1

Usability of interactive System-Introduction, Usability goals and measure, Usability Motivations.

Managing Design Processes-The four pillars of Design, Development Methodologies.

Module 2

Direct manipulation and Virtual Environments- Introduction, Examples of Direct Manipulation, 3-D Interfaces, Teleoperation, Virtual and Augmented Reality.

Menu Selection, Form Fill-in and Dialog Boxes-Single Menus, Data Entry with Menus: Form Fill-In, Dialog Boxes and Alternatives.

Module 3

Essential PHP-Creating a first PHP Page, Mixing HTML and PHP, Printing some text, Printing some HTML, Adding comments to PHP code, Working with variables, Storing Data in variables, Creating Constants, Understanding PHP's Internal Data Types.

Operators- PHP Arithmetic Operators, Working with Assignment Operator, Comparison Operator, Increment/Decrement Operator, Logical Operator, String Operator, Bitwise Operator, Comparison Operator, Ternary Operator

Module 4

PHP Conditional Statements-The if statement, the else statement, the elseif statement, the switch statement, for loops, while loops, do while loops, foreach loop, terminating loops(break/continue).

Strings and Arrays-The String functions, converting to and from strings, creating an array, Modifying the data in Arrays, Deleting Array Elements, Handling array with Loops, PHP Array Functions, converting between strings and arrays using implode and explode, Sorting Arrays, Using PHP's Array Operators.

Functions-Creating Functions in php, passing functions some data, passing arrays to Functions, Passing by References, using default Arguments, returning data from functions, Returning Arrays, Introducing Variable Scope in PHP, Accessing Global data, Working with Static Variables, Creating include files.

Module 5

PHP Forms-Reading data in Web Pages (Get and Post), handling text fields, handling text areas, handling Check boxes, handling Radio Buttons, handling Password Controls, handling hidden controls, handling file uploads.

Web features - Sessions-Storing data in Sessions, retrieving data from sessions, destroy sessions, Cookies-Setting a cookie, reading a Cookie, Setting Cookies' Expiration, deleting Cookies.

Accessing The Database in PHP (Using MySQL/MySQLi procedural):-Connecting to the database server, Connecting to the Database, Displaying the table data, Closing the Connection, Updating database, Inserting new data items into a database, delete records, Creating new tables, creating new Database.

Text Books

1. Ben Shneiderman, Catherine Palisant, Maxine Cohen, Steven Jacobs *"Designing the user Interface, 5th edition"*
2. Steven Holzner "The Complete Reference PHP, Mc Graw Hill Education"

Reference Books

1. Benyon, D. (2019). *Designing User Experience: A guide to HCI, UX and interaction design* (4th ed.). Pearson Education.
2. Nixon, R. (2021). *Learning PHP, MySQL & JavaScript: A Step-by-Step Guide to Creating Dynamic Websites* (6th ed.). O'Reilly Media

Web Resources

1. https://www.w3schools.com/php/default.asp
2. https://www.tutorialspoint.com/php/index.htm

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Module 1**	**6 hrs.**
1.1	Usability of interactive System	3
1.2	Managing Design Processes	3
2	**Module 2**	**9 hrs.**
2.1	Direct manipulation and Virtual Environment	5
2.2	Menu Selection, Form Fill-in and Dialog Boxes	4
3	**Module 3**	**6 hrs.**
2.1	Essential PHP	3
2.2	Operators	3
4	**Module 4**	**12 hrs.**
4.1	Conditional Statements	3
4.2	Strings	3
4.3	Arrays	3
4.4	Functions	3
5	**Module 5**	**15 hrs.**
5.1	PHP Forms	3
5.2	Sessions	3
5.3	Cookies	3
5.4	Accessing the database in PHP	6

24SJINMCA305	INTRODUCTION TO RDBMS AND SQL	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course provides a clear understanding of fundamental principles of Database Management Systems (DBMS) with special focus on Relational Databases to the learners. The topics covered in this course are basic concepts of DBMS, Entity Relationship (ER) model, Relational Database principles, Structured Query Language (SQL), and Normalization.
Prerequisite:	NIL

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Summarize and exemplify fundamental nature and characteristics of database systems	K2
CO2	Conceptualize data using the relational model and describe relational database concepts	K3
CO3	Develop queries for relational database in the context of practical applications	K4
CO4	Model and design relational databases following the design principles	K4
CO5	Solve database problems using SQL and PLSQL.	K3

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	1	2	1	1	1	2
CO2	3	3	2	2	1	1	1	2
CO3	3	3	3	3	2	1	1	2
CO4	3	3	3	3	2	2	1	2
CO5	3	3	3	3	2	2	1	3

Mark distribution:

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. What is data independence, and why is it lacking in file systems?
2. What is a DBMS, and what are its functions?

Course Outcome 2 (CO2):

1. Why are entity integrity and referential integrity important in a database?
2. Discuss two ways in which the 1:M relationship between COURSE and CLASS can be implemented.

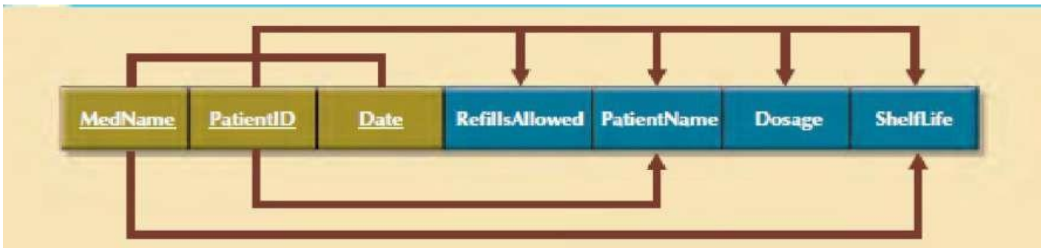
Course Outcome 3(CO3):

1. Rewrite the following WHERE clause without the use of the IN special operator. WHERESTATE IN ('TN', 'KL', 'KA')
2. Write the SQL code required to list all employees whose last names start with Smith. In other words, the rows for both *Smith* and *Smithfield* should be included in the listing. Assume case sensitivity.

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INTIAL	EMP_HIREDATE	JOB_CODE
101	Smith	John	G	08-Nov-00	502
102	Senior	David	H	12-Jul-89	501
103	Arbough	June	E	01-Dec-96	500
104	Ramoras	Annae	K	15-Nov-87	501
105	Johnson	Alice	K	01-Feb-93	502
106	Smithfield	William		22-Jun-04	500
107	Alonzo	Maria	D	10-Oct-93	500

Course Outcome 4 (CO4):

1. What is a partial dependency? With what normal form is it associated?
2. The dependency diagram in below figure indicates that a patient can receive many prescriptions for one or more medicines over time. Based on the dependency diagram, create a database whose tables are in at least 2NF, showing the dependency diagram for each table.



Course Outcome 5 (CO5):

1. The relational set operators UNION, INTERSECT, and MINUS work properly only when the relations are union-compatible. What does union-compatible mean, and how would you check for this condition?
2. Why does the order of the operands(tables) matter in a MINUS query but not in a UNION query?

Syllabus

Module 1

Database Management System Concepts: Introduction, Data, information, Introducing the Database, Importance of Database Design, Evolution of File System Data Processing, Problems with file system data processing, Database Systems- The database system environment, DBMS Functions. Degrees of Data Abstraction

Data Models- Data Modeling and Data Models, Entity Relationship (ER) Modeling – The entity Relationship Model, Developing an ER Diagram, Extended ER Features: Specialization, Generalization, Aggregation.

Module 2

Introduction to the Relational Database Model: Structure of relational database, Degree and Cardinality, Database Schema, Schema Diagrams, Keys, Integrity constraints, Indexes, Codds Relational Database Rules. Relationships within the Relational Database, Difference between DBMS and RDBMS

Module 3

Introduction to SQL: Introduction to SQL, Data Definition Commands, Data Manipulation Commands, Transaction Control Commands, Data Control Commands, Null Values, SELECT Queries, Additional Data Definition Commands, Additional SELECT Query Keywords, Group By and Having Clause, Aggregate Functions, Numeric Functions, String Functions, Date Functions.

Module 4

Normalization: Functional Dependency, Types of Functional Dependency, Armstrong's Axioms (proofs not required), Closures and their computation, Equivalence of Functional Dependencies (FD), The need for Normalization, First Normalization, Second Normal Form

Relation, Third Normal Form, Boyce-Codd Normal Form (BNCF), Fourth and Fifth Normal Form.

Module 5

Advanced SQL: Relational Set Operators, Joining Database Tables. Different types of joins, Subqueries and Correlated Queries, Views, Updatable Views, Functions and Procedures, Cursors, Triggers.

Text Books

1. Peter Rob and Carlos Coronel, "Database Principles: Fundamentals of Design, Implementation and Management", 9th Edition(2012)
2. Henry Korth and A. Silberschatz, "Database System Concepts", Sixth Edition, McGraw- Hill,2011
3. Ramez Elmasri & Shamkant B. Navathe, "Fundamentals of Database Systems",7th Edition, Pearson Education (2016).
4. Ashutosh Kumar Dubay, "Database Management Concepts", S.K. Kataria & Sons, 1st Edition (2012).
5. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", McGraw Hill, 3rd Edition (2014).

Web Resources

1. Introduction to Databases (NPTEL):https://onlinecourses.nptel.ac.in/noc20_cs03/preview
2. Database Systems Concepts & Design: https://www.udacity.com/course/sql-for-data-analysis--ud198

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Module 1**	**9 hrs.**
1.1	Introduction, Data, information, Introducing the Database, Importance of Database Design	1
1.2	Evolution of File System Data Processing, Problems with file system data processing	1
1.3	Database Systems- The database system environment	1
1.4	DBMS Functions	1
1.5	Degrees of Data Abstraction	1
1.6	Data Models- Data Modeling and Data Models	1

1.7	Entity Relationship (ER) Modeling – The entity Relationship Model	1
1.8	Developing an ER Diagram	1
1.9	Extended ER Features: Specialization, Generalization, Aggregation.	1
2	**Module 2**	**9 hrs.**
2.1	Structure of relational database	1
2.2	Degree and Cardinality	1
2.3	Database Schema, Schema Diagrams	1
2.4	Keys, Integrity Constraints	2
2.5	Indexes, Codds Relational Database Rules.	2
2.6	Relationships within the Relational Database, Difference between DBMS and RDBMS	2
3	**Module 3**	**12 hrs.**
3.1	Introduction to SQL, Data Definition Commands	1
3.2	Data Manipulation Commands	2
3.3	Data Control Commands, Null Values	1
3.4	SELECT Queries	2
3.5	Additional Data Definition Commands	1
3.6	Additional SELECT Query Keywords	1
3.7	Group by and Having Clause	2
3.8	Aggregate Functions, Numeric functions, String functions, Date functions	2
4	**Module 4**	**9 hrs.**
4.1	Functional Dependency, Types of Functional Dependency	1
4.2	Armstrong's Axioms (proofs not required)	1
4.3	Closures and their computation, Equivalence of Functional Dependencies (FD)	1
4.4	The need for Normalization, First Normalization	1
4.5	Second Normal Form Relation	1
4.6	Third Normal Form	1
4.7	Boyce-Codd Normal Form (BNCF)	1
4.8	Fourth and Fifth Normal Form	2
5	**Module 5**	**9 hrs.**
5.1	Relational Set Operators	1
5.2	Joining Database Tables	1
5.3	Different types of joins	1
5.4	Subqueries and Correlated Queries	2
5.5	Views, Updatable Views	1
5.6	Functions and Procedures	1
5.7	Cursors	1
5.8	Triggers	1

24SJINMCA307	FUNDAMENTALS OF INFORMATION SYSTEMS SECURITY	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course provides a solid theoretical foundation of information security to protect information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction.
Prerequisite:	24SJINMCA210 Internet Concepts and Web Technology, 24SJINMCA206 Operating Systems.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand and explain fundamental security concepts	K2
CO2	Recognize common threats and vulnerabilities of information systems.	K2
CO3	Identify and analyze the Risk Management process.	K3
CO4	Explain and apply cryptographic algorithms.	K3
CO5	Understand and apply network security enumeration and monitoring tools.	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2					1	2
CO2	2	3					3	2
CO3	2	3				2	2	2
CO4	3	2	3	3			2	2
CO5	2	3	3	3			2	2

Mark Distribution

Total Marks	CIA	ESE	ESE Duration
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. List out the components of an information system.
2. Describe the goals of information security.
3. Summarize Security in the Systems Life Cycle.

Course Outcome 2 (CO2):

1. Explain Espionage or Trespass.
2. Illustrate alternative approaches for password cracking.
3. Categorize Social Engineering Attacks.

Course Outcome 3 (CO3):

1. Explain security and privacy issues in Cloud Computing.
2. With necessary diagram, demonstrate components of risk management.
3. Compose steps for protecting your personal computer.

Course Outcome 4 (CO4):

1. Outline Firewall processing modes.
2. Diagrammatically explain digital signature.
3. Compare HTTP and HTTPS protocol.

Course Outcome 5 (CO5):

1. Differentiate physical security controls.
2. Outline Security Management Maintenance Models.
3. Explain the methods of data interception.

Syllabus

Module 1

Introduction to Information System Security: History, Key Concept in Information Security, Information and Characteristics, Goals of information security, Need for Security, Components, Balancing security and access, Implementation issues of the goals of information security, Security in the Systems Life Cycle, Security professional in the organization.

Module 2

The Need for Security: Business Needs – Protecting the functionality, Enabling the safe operations, Protecting the data, safe guarding the technology assets, Threats and Attacks: Threats - compromises to Intellectual property, deliberate software attacks, Espionage and trespass, sabotage and vandalism, Attacks-Malicious Codes, Back Doors, Denial of Service and Distributed Denial of Service, Spoofing, sniffing, Spam, Social Engineering, Theft.

Module 3

Risk Management: Definition, Risk Identification, Risk Assessment, Risk Control. Information Technology in Today's World: Protecting your Personal Computer, Challenges, attacks, Security measures, Securing Mobile and Portable Systems, Cloud Computing, Deployment, Security and Privacy Issues, Trust and Risk.

Module 4

Security Technology: Access Control, password Security, Firewalls, Protecting Remote Connections, Intrusion Detection and Prevention Systems, Honeypots, Honeynets, and Padded Cell Systems, Scanning and Analysis Tools.

Cryptography: Cipher Methods, Cryptographic Algorithms, Cryptographic Tools, Public Key Infrastructure (PKI), Digital Signatures, Digital Certificates, Protocols for Secure Communications- S-HTTP, S/MIME, SET, and SSH, WPA, IPSec, PGP.

Module 5

Physical Security, Implementation and Maintenance: Physical Access Controls, Interception of Data, Implementing Information security: Information Security Project Management, Technical Aspects of Implementation, Nontechnical Aspects of Implementation, Information Systems Security Certification and Accreditation. Information Security Maintenance: Security Management Maintenance Models, Digital Forensics.

Text Books

1. Whitman, M. E., & Mattord, H. J. (2014). *Principles of Information Security* (5th ed.). Cengage Learning.

Reference Books

1. Peltier, T. R. (2010). *Information Security Risk Analysis* (3rd ed.). Auerbach Publications.
2. Stamp, M. (2011). *Information Security: Principles and Practice* (2nd ed.). John Wiley & Sons.
3. Conklin, W. A., White, G. B., Cothren, C., Davis, R. L., & Williams, D. (2018). *Principles of Computer Security: CompTIA Security+ and Beyond* (5th ed.). McGraw-Hill Education.
4. Stallings, W. (2013). *Cryptography and Network Security: Principles and Practice* (6th ed.). Pearson Education.

Web Resources

1. https://www.geeksforgeeks.org/what-is-information-security/

2. https://en.wikibooks.org/wiki/Fundamentals_of_Information_Systems_Security/Information_Security_and_Risk_Management
3. https://pressbooks.pub/bus206/chapter/chapter-1/

Course Contents and Lecture Schedule

No.	Topic	No. of lectures
1	**Module 1**	**10 hrs.**
1.1	Introduction, History, Key Concept in Information Security	1
1.2	Information and Characteristics, Goals of information security	2
1.3	Need for Security, Components	2
1.4	Balancing security and access, Implementation issues of the goals of information security	2
1.5	Security in the Systems Life Cycle	2
1.6	Security professional in the organization	1
2	**Module 2**	**8 hrs.**
2.1	Business Needs-Protecting the functionality, Enabling the safe operations, Protecting the data, safe guarding the technology assets	2
2.2	Threats and Attacks: Threats-compromises to Intellectual property, deliberate software attacks, Espionage and trespass	2
2.3	sabotage and vandalism, Attacks-Malicious Codes, Back Doors, Denial of Service and Distributed Denial of Service	2
2.4	Spoofing, sniffing, Spam, Social Engineering, Theft	2
3	**Module 3**	**6 hrs.**
3.1	Risk Management: Definition, Risk Identification, Risk Assessment, Risk Control	2
3.2	Protecting your Personal Computer, Challenges, attacks, Security measures	1
3.3	Securing Mobile and Portable Systems	1
3.4	Cloud Computing, Deployment, Security and Privacy Issues, Trust and Risk	2
4	**Module 4**	**16 hrs.**
4.1	Security Technology: Access Control, password Security	1
4.2	Firewalls, Protecting Remote Connections, Intrusion Detection and Prevention Systems	2
4.3	Honeypots, Honeynets, and Padded Cell Systems, Scanning and Analysis Tools	1
4.4	Cryptography: Cipher Methods	2
4.5	Cryptography: Algorithms	2
4.6	Cryptographic Tools	3
4.7	Public Key Infrastructure (PKI), Digital Signatures	2
4.8	Digital Certificates	1
4.9	Protocols for Secure Communications- S-HTTP, S/MIME, SET, and SSH, WPA, IPSec, PGP	2

5	**Module 5**	**8 hrs.**
5.1	Physical Security: Physical Access Controls, Interception of Data	2
5.2	Implementing information security: Information Security Project Management	1
5.3	Technical Aspects of Implementation, Nontechnical Aspects of Implementation	1
5.4	Information Systems Security Certification and Accreditation	1
5.5	Information Security Maintenance: Security Management Maintenance Models	2
5.6	Digital Forensics	1

24SJINMCA309	INTRODUCTION TO OPERATIONS RESEARCH	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	Operations research (OR) is an analytical method of problem-solving and decision-making. The topics Linear Programming, Transportation and Assignment problems, game theory and queuing models are introduced here.
Prerequisite:	A basic course in matrix algebra and statistics.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Model and solve decision making problems using Linear Programming techniques	K4
CO2	Solve linear programming problems using artificial variables and duality.	K5
CO3	Apply the concepts of transportation and assignment problem to solve different practical problems.	K5
CO4	Use game theory to solve real world problems.	K4
CO5	Understand and extend queuing models to analyse real world systems.	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	1	2		2
CO2	3	3	2	3	1	2		2
CO3	3	3	3	3	1	3		2
CO4	3	3	3	2	1	2		2
CO5	3	3	3	3	1	2		3

Mark distribution:

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define Slack variables and Surplus variable.
2. Solve by Graphical method: Maximise Z = $5x_1 + 3x_2$
   subject to $x_1 + x_2 \leq 6$, $2x_1 + 3x_2 \geq 6$, $0 \leq x_1 \leq 4$, $0 \leq x_2 \leq 3$.
3. Solve by Simplex method: Maximise Z = $5x_1 + 3x_2$
   subject to $x_1 + x_2 \leq 6$, $2x_1 + 3x_2 \geq 6$, $0 \leq x_1 \leq 4$, $0 \leq x_2 \leq 3$.

Course Outcome 2 (CO2):

1. Solve by Big M method
   Maximise Z = $3x_1 + 2x_2 + 3x_3$
   Subject to $2x_1 + x_2 + x_3 \leq 2$;
   $3x_1 + 4x_2 + 2x_3 \geq 8$
   $x_1, x_2, x_3 \geq 0$.
2. Use Two Phase method to
   Maximise Z = $3x_1 - x_2$
   Subject to $2x_1 + x_2 \geq 2$,
   $x_1 + 3x_2 \leq 2$
   $x_2 \leq 4$
   $x_1, x_2 \geq 0$.
3. Evaluate the dual and solve
   Minimize Z = $15x_1 + 10x_2$
   Subject to $3x_1 + 5x_2 \geq 5$,
   $5x_1 + 2x_2 \geq 3$,
   $x_1, x_2 \geq 0$.

Course Outcome 3 (CO3):

1. Find an initial basic feasible solution by Least Cost method:

	A	B	C	Availability
1	6	3	5	10
2	8	2	7	15
3	4	9	1	8
Requirement	15	7	11	

2. Find the optimal solution of the following Transportation problem

Market

Plant	A	B	C	D	Available
X	14	9	18	6	11
Y	10	11	7	16	13
Z	25	20	11	34	19
Requirement	6	10	12	15	

3. There are five jobs to be assigned, one each to five machines, and the associated cost matrix as follows. Select the optimum assignment of machines to jobs.

Job Machine

	1	2	3	4	5
A	11	17	8	16	20
B	9	7	12	6	15
C	13	16	15	12	16
D	21	24	17	28	26
E	14	10	12	11	15

Course Outcome 4 (CO4):

1. Define Maximin and Minimax Principle.
2. Apply dominance rule and solve

   8 10 9 14

   [10 11 8 12]

   13 12 14 13
3. Solve the following problem graphically

   1 −3

   3 5

   −1 6

   4 1

   2 2

   −5 0

Course Outcome 5 (CO5):

1. Classify different queuing models.
2. A T.V repairman finds that the time spent on his jobs has an exponential distribution with mean 30 minutes. If he repairs sets in the order in which they came in, and if the arrival of sets is approximately Poisson with an average rate of 10 per 8 - hour day,

what is repairman's expected idle time each day? How many jobs are ahead of the average set just brought in?

3. On an average 96 patients per 24-hour day require the service of an emergency clinic. Also, on an average a patient requires 10 minutes of active attention. Assume that the facility can handle only one emergency at a time. Suppose that it costs the clinic Rs 100 per patient treated to obtain an average servicing time of 10 minutes, and that each minute of decrease in this average time would cost Rs 10 per patient treated. How much would have to be budgeted by the clinic to decrease the average size of the queue from $1\frac{1}{3}$ patients to $\frac{1}{2}$ a patient.

Syllabus

Module 1

Linear Programming Problem-Formulation-Graphical Method-Simplex method. (Theorems without proof)

Module 2

Big M Method-Two Phase Method-Duality-Solving Linear Programming Problem using duality. (Theorems without proof)

Module 3

Transportation problem -Methods to find initial basic feasible solution –Northwest corner rule-Matrix minima method-Vogel's Approximation method. Solving a Transportation problem -MODI method - Degeneracy in Transportation problem -Unbalanced Transportation problem -Maximization in Transportation problem. Assignment problem -Hungarian Method of Assignment-Unbalanced Assignment Problem-Maximization case in Assignment Problem-Prohibited Assignments.

Module 4 (Curve Fitting)

Game Theory- Two person zero sum game -Basic terms-saddle point-Maximin- Minimax principle- Games without saddle points -Mixed strategies -Algebraic method for solving two person zero sum game-Graphical solution of 2xn and mx2 games-Dominance principle-Solving mxn game using dominance.

Module 5

Queuing Theory- Elements of a queuing system- Kendall 's notation-Operating characteristics. Poisson process, Exponential distribution, Birth and death process (All without proof). Queuing models based on Poisson process- Single server models with finite and infinite capacity-Multi sewer models with finite and infinite capacity. (Theorems without proof).

Textbooks

1. Gupta, P. K., & Hira, D. S. (2022). *Introduction to operations research*. S. Chand & Company
2. Swarup, K., Gupta, P. K., & Mohan, M. (2022). *Operations research* (20^{th} ed.). Sultan Chand & Sons.
3. Taha, H. A. (2022). *Operations research: An introduction* (11^{th} ed.). Pearson Education.
4. Ravindran, A., Phillips, D. T., & Solberg, J. J. (Year unknown). *Operations research: Principles and practice* (2nd ed.). Wiley India.

Reference Text Books

1. G Hadley, Linear programming, Narosa Publishing House, New Delhi,2002
2. Hillier & Lieberman, Introduction to Operations Research, Holden Day Inc.,1996
3. R Panneerselvam, Operation Research. PHI, 2006

Web Resources

1. https://onlinecourses.nptel.ac.in/noc22_ma48/preview
2. https://onlinecourses.nptel.ac.in/noc24_mg30/preview

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Module 1**	**9 hours**
1.1	Course Contents and Lecture Schedule	1
1.2	Formulation	2
1.3	Graphical Method	3
1.4	Simplex method	3
2	**Module 2**	**9 hours**
2.1	Big M Method	3
2.2	Two Phase Method	3
2.3	Duality-Solving LPP using duality	3
3	**Module 3**	
3.1	Transportation problem -Methods to find initial basic feasible solution	1

3.2	Northwest corner rule- Matrix minima method	1
3.3	Vogel's Approximation method	1
3.4	Solving a TP -MODI method	2
3.5	Degeneracy in TP	1
3.6	Unbalanced TP-Maximization in TP	1
3.7	Assignment problem -Hungarian Method of Assignment	3
3.8	Maximization in assignment problem	1
4	**Module 4**	**9 hours**
4.1	Game Theory: Two-person zero sum game -Basic notions-saddle point	1
4.2	Maximin-Minimax principle- Games without saddle point - Mixed strategies	2
4.3	Algebraic method for solving two-person zero sum game	2
4.4	Graphical method for 2xn and mx2 games	2
4.5	Dominance Principle-Solving mxn game -using dominance	2
5	**Module 5**	**10 hours**
5.1	Queuing Theory- Elements of a queuing system- Kendall 's notation- Operating characteristics	1
5.2	Poisson process, Exponential distribution, Birth and death process (All without proof).	3
5.3	Queuing models based on Poisson process- Single server models with finite and infinite capacity	3
5.4	Multi server models with finite and infinite capacity	3

24SJINMCA331	RDBMS LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	0	4	1

Preamble:	The syllabus is prepared with the view of preparing the MCA Graduates capable of writing SQL, PL/SQL to solve computational problems that they may have to solve in their professional life. The students can explore various data manipulations for the basic problem solving. The course content is decided to cover the essential RDBMS fundamentals. The students are expected to come prepared with the required program written in the rough record for the lab classes.
Prerequisite:	24SJINMCA305 Introduction to RDBMS and SQL.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand, appreciate and effectively explain the underlying concepts of database technologies	K2
CO2	Design and implement a database schema for a given problem-domain.	K5
CO3	Apply different normalization techniques	K4
CO4	Populate and query a database using SQL DML/DDL commands.	K5
CO5	Use any popular RDBMS for data access and updating.	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	3	2		2	2	2	3
CO2	2	2	3	3	2	3		
CO3	3	3	3	3	3	3		
CO4	3	3	3	3	3	3	2	
CO5	3	3	3		2	2	3	2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	7½
Maintenance of daily lab record and GitHub management	10
Regular class viva voce	7½
Timely completion of day-to-day tasks	10
Tests/Evaluation	15

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 marks</td></tr>
<tr><td rowspan="4">Problem solving
(Based on
or more questions may be given)</td><td>Flowchart / Algorithm / Structured
description of problem to explain how the</td><td>15%</td><td rowspan="4"></td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output</td><td>20%</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explore the commands to retrieve the contents of the table.
2. Analyse the structure of a database object.

Course Outcome 2 (CO2)

1. DDL commands for table creation.
2. SQL to create a view.

Course Outcome 3(CO3):

1. Select/Display the Student name and grade by implementing a left outer join.
2. Select/Display the Student name, register no, and result by implementing a right outer

join.

3. List the records in the emp table order by salary in ascending order.

Course Outcome 4 (CO4):

1. Grade setting for a student using trigger.
2. Categorize the grades of students using cursor.

Course Outcome 5 (CO5):

1. Design and implement payroll management system.
2. Demonstrate and implement banking management system.

Reference Books

1. Peter Rob and Carlos Coronel, "*Database Principles: Fundamentals of Design, Implementation and Management*", 9th Edition (2012)
2. Ashutosh Kumar Dubay, "*Database Management Concepts*", S.K. Kataria & Sons, 1st Edition (2012).
3. Raghu Ramakrishnan and Johannes Gehrke, "*Database Management Systems*", McGraw Hill, 3rd Edition (2014).
4. Henry Korth and A. Silberschatz, "*Database System Concepts*", Sixth Edition, McGraw-Hill,2011.

Web Resources

1. Introduction to Databases (NPTEL): https://onlinecourses.nptel.ac.in/noc20_cs03/preview
2. Database Systems Concepts & Design https://www.udacity.com/course/databasesystems-concepts-design--ud150

Reference List of Lab Experiments

1. Create a table called EMP with the following structure:

   EMPNO NUMBER (6)

   ENAME VARCHAR2(20)

   JOB VARCHAR2(10)

DEPTNO NUMBER (3)

SAL NUMBER (7,2)

Allow NULL for all columns except ename and job.

2. Add a column experience to the emptable.
3. Modify the column width of the job field of emptable.
4. Create dept table with the following structure.

   DEPTNO NUMBER (2)

   DNAME VARCHAR2(10)

   LOC VARCHAR2(10)

   DEPTNO as the primary key
5. Create the emp1 table with ename and empno, add constraints to check the empno value while entering (i.e.) empno >100.
6. Drop a column experience to the emptable.
7. Insert a single record into dept table.
8. The organization wants to display only the details of the employees those who are ASP.
9. The organization wants to display only the details like empno, empname, deptno, deptname of the employees. (Vertical portioning)
10. Display all the views generated
11. Drop a view.
12. Insert more than a record into emp table using a single insert command.
13. Update the emp table to set the salary of all employees to Rs15000/- who are working as ASP.
14. Create a pseudo table employee with the same structure as the table emp and insert rows into the table using select clauses.
15. Select employee name, job from the emp table.
16. Delete only those who are working as lecturer.
17. List the records in the emp table order by salary in ascending order.
18. List the records in the emp table order by salary in descending order.
19. Display deptno from the table employee avoiding the duplicated values.
20. Site examples for the use of data and built-in functions in SQL.
21. Site examples for the use of date & time functions.
22. Site examples for the use of Numeric & Math functions.

23. Display all employee names and salary whose salary is greater than minimum salary of the company and job title starts with 'M '.
24. Issue a query to find all the employees who work in the same job as Arjun.
25. Display all the dept numbers available with the dept and emp tables avoiding duplicates.
26. Display all the dept numbers available with the dept and emp tables.
27. Display all the dept numbers available in emp and not in dept tables and vice versa.
28. Display the employee details, departments that the departments are same in both the emp and dept.
29. Display the employee details, departments that the departments are not same in both the emp and dept. Solution: 1. Use select from clause. 2. Use non equi join in select clause to get the result.
30. Display the Student name and grade by implementing a left outer join.
31. Display the Student name, register no, and result by implementing a right outer join.
32. Display the Student name register no by implementing a full outer join.
33. Write a query to display their employee names.
34. Display the details of those who draw the salary greater than the average salary.
35. Develop a query to grant all privileges of employees table into departments table.
36. Develop a query to grant some privileges of employees table into departments table.
37. Develop a query to revoke all privileges of employees table from departments table.
38. Develop a query to revoke some privileges of employees table from departments table.
39. Write a query to implement the save point.
40. Write a query to implement the rollback.
41. Write a query to implement the commit.
42. Write a program to implement cursor for setting grades for a student group.
43. Use trigger for setting the grade for the student.
44. Create a function to find the factorial of a number.
45. Implementation of embedded SQL for finding the BONUS.
46. Design database and implement payroll processing.
47. Design and implementation of banking system.
48. Design and implementation of library management system.
49. Create backup for database.
50. How can you restore the database?

Course Contents and Lecture Schedule

No.	Topic	No. of Hours
1	**Module 1**	**12 hrs.**
1.1	Table Design- Using foreign key and Normalization	2
1.2	Practice SQL Data Definition Language (DDL) commands • Table creation and alteration (include integrity constraints such as primary key, referential integrity constraints, check, unique and null constraints both column and table level • Other database objects such as view, index, cluster, sequence, synonym etc.	10
2	**Module 2**	**12 hrs.**
2.1	Practice SQL Data Manipulation Language (DML) commands 1. Row insertion, deletion and updating 2. Retrieval of data 3. Simple select query 4. Select with where options (include all relational and logical operators) 5. Functions: Numeric, Data, Character, Conversion and Group functions with having clause. 6. Set operators 7. Sorting data 8. Sub query (returning single row, multiple rows, more than one column, correlated subquery) 9. Joining tables (single join, self-join, outer join)	12
3	**Module 3**	**6 hrs.**
3.1	Practice Transaction Control Language (TCL) commands (Grant, revoke, commit and save point options)	6
4	**Module 4**	**6 hrs.**
4.1	Usage of triggers, functions and procedures using PL/SQL constructs.	6
5	**Module 5**	**12 hrs.**
5.1	Development of sample applications using Oracles Back End Sample applications may include 1. Payroll Information System 2. Student Information System 3. Bank transaction 4. Library Information System etc.	10
5.2	How to take back up and restore using Oracle	2

24SJINMCA333	**USER INTERFACE DESIGN LAB**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		PRACTICAL	0	0	4	1

Preamble:	This course challenges the students how to make your pages dynamic based upon user interaction, interacting with HTML forms and store and retrieve information from local data sources which include a database.
Prerequisite:	24SJINMCA305 Introduction to PHP and MySQL

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand how server-side programming works on the web.	K5
CO2	Creating conditional structures, Storing data in arrays and strings	K5
CO3	Develop PHP Programs with functions.	K5
CO4	Understanding POST and GET in form submission.	K4
CO5	Create a database in phpMyAdmin and develop a web application.	K5

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	2	3	1			2
CO2	3	3	2	3	1			2
CO3	3	3	3	3	2			2
CO4	3	2	2	3	1			2
CO5	3	3	3	3	2	2	1	3

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	7½
Maintenance of daily lab record and GitHub management	10
Regular class viva voce	7½

Timely completion of day-to-day tasks	10
Tests/Evaluation	15

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 marks</td></tr>
<tr><td rowspan="4">Problem solving (Based on difficulty level, one or more questions may be given)</td><td>Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output</td><td>20%</td></tr>
</table>

Course Level Assessment Questions

Course Outcome (C01):

1. Evaluate value of Z where Z= $(x^2 + y3)/24x$.
2. Develop a PHP program to swap the values without a third variable.

Course Outcome 2 (C02):

1. Develop a menu driven program for performing to perform arithmetic operations: a) Addition b) Subtraction c) Multiplication d) Division using Switch statement.
2. Develop a PHP Program to find the largest element in an array.

Course Outcome 3 (C03):

1. Develop a PHP program to include another file in PHP.
2. Design a PHP program to find the factorial of a number.

Course Outcome 4 (C04):

1. Outline a registration form using PHP and do necessary validations.
2. Model a login and log out form having username and password and display the name in dashboard using Session.

Course Outcome 5 (C05):

1. Build a PHP program to implement database connectivity.
2. Generate a user profile.

Syllabus

Server-Side Scripting using PHP: Setting up the environment (Example-XAMP server), PHP Programming basics – Print/echo, Variables and constants, Operators, Control structures and looping structures, Strings and Arrays, Functions, PHP include and require Statements,

Embedding PHP within HTML, Reading Data in Web Pages (GET and POST), Form Validation, Sessions, establishing connectivity with database, Implement Web Application using PHP and MySQL, File upload.

Reference Books

1. Steven Holzner "The Complete Reference PHP, Mc Graw HillEducation
2. Steve Suehiring,Tim Converse and Joyce Park "PHP6 and MysqlBible",Wiley

Web Resources

1. https://training.trainingtrains.com/php-tutorial.html
2. https://www.geeksforgeeks.org/php/php-examples/
3. Create MySQL Database Using PHP

List of Lab Experiments

1. Read 3 integer values and find the largest among them.
2. Read a Natural Number and check whether the number is prime or not.
3. Read a Natural Number and check whether the number is Armstrong or not.
4. Read a Natural Number and find the reverse of the number.
5. Read 2 integers values and swap the values without a third variable
6. Convert temperature from Fahrenheit to degree Celsius.
7. Read a natural number and find the sum of digits.
8. Develop a menu driven program for performing to perform arithmetic operations:

a) Addition b) Subtraction c) Multiplication d) Division using Switch statement.

9. Find Fibonacci series up to n.
10. Find the largest and smallest element in an array.
11. Develop a PHP program to display the odd and even numbers in a given set of numbers.
12. Check the given matrix is symmetric or not.
13. Find the length of a string using string function.
14. Find the reverse of a string using string function.
15. Read 2 integers values and swap the values using function.
16. Find factorial of a number using function.
17. Develop a PHP program to include PHP file in HTML.
18. Develop a registration form using PHP and do necessary validations (Name, Email, DOB, Mobile number)
19. Develop a login and log out form having username and password and display the name in dashboard using Session.
20. Develop a PHP program to implement database connectivity.
21. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and perform following Operations a) Insert the records b) Update the records d) delete the record d) search for a book with the title specified by the user e) display the search results with proper headings.
22. Using PHP and MySQL, develop a user profile to accept Name, Profile Picture, Address and Phone number from a web page and store the information in a database and display it in neat format.

Course Contents and Lecture Schedule

No.	Topic	No. of Hours
1	**Module 1**	**6 hrs.**
1.1	Setting up the environment	2
1.2	PHP Programming basics	4
2	**Module 2**	**12 hrs.**
2.1	Control structures and looping structures	6
2.2	Arrays and String	6
3	**Module 3**	**6 hrs.**
3.1	Functions	3

3.2	PHP include and require Statements	3
4	**Module 4**	**12 hrs.**
4.1	Embedding PHP within HTML	3
4.2	Reading Data in Web Pages (GET and POST)	3
4.3	Form validation	3
4.4	Sessions	3
5	**Module 5**	**12 hrs.**
5.1	Establishing connectivity with database.	4
5.2	Implement Web application using PHP and Mysql	4
5.3	File Upload	4

ST.JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

MCA Integrated

SEMESTER 6

24SJINMCA302	INTRODUCTION TO SOFTWARE ENGINEERING	CATEGORY	L	T	P	CREDIT
		CORE	3	1		4

Preamble	This course introduces students to the basic theory of software engineering, and to the applications of these basic theoretical principles to a software development project.
Prerequisite	Basic knowledge of programming, computer fundamentals, problem-solving, and mathematics.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the different development practices and its advantages	K3
CO2	Understand the different process models and choose the best model for their project	K3
CO3	Understand and meet ethical standards and legal responsibilities.	K4
CO4	Work as an effective member or leader of software engineering teams.	K3
CO5	Understand the environment and work culture in a software organization	K3

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	2	1				2
CO2	3	3	2	2		2		2
CO3							3	2
CO4		2	2		3	2		2
CO5					2	2	2	3

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 Marks
Continuous Assessment Test (2 numbers)	: 20 Marks
Assignment/Quiz/Course project	: 12 Marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer anyone. Each question can have a maximum 2 subdivisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Summarize the various approaches for "Recording" the Requirements.
2. The high-level design should sketch out all the application's security needs. Evaluate the methods by which we can ensure security for our application
3. Imagine you are in a project team and the Team Leader asked you to sketch out an UML diagram for a particular scenario so that the diagram should focus on a sequence of messages. Which will be the best UML diagram you can draw in this particular scenario? Justify your answer with an example.

Course Outcome 2 (CO2):

1. Compare the waterfall with feedback model with the sashimi model?
2. The waterfall model and the prototyping model can be accommodated in the Spiral process model. Justify
3. Demonstrate any one project size estimation technique and its effort calculation using COCOMO with example.

Course Outcome 3(CO3):

1. Illustrate the agile process? List out its characteristics.
2. Justify the importance of pair programming?
3. Illustrate Refactoring? Justify its importance. List out different Refactoring techniques.

Course Outcome 4 (CO4):

1. Describe the characteristics of user stories.
2. Explain Scrum framework.
3. Illustrate the various tools for Agile project management.

Course Outcome 5 (CO5):

1. Demonstrate the importance of CI/CD
2. DevOps is a set of practices that works to automate and integrate the processes between software development and IT teams. Justify.
3. State the benefits of release monitoring

SYLLABUS

Module 1

Introduction to Software Engineering: What is Software Engineering - Why is software engineering important, Details around requirements gathering, Software design, Development, Testing, Deployment, Maintenance. Planning phase – project planning objective, software scope, empirical estimation, models, COCOMO, staffing and personal planning.

Module 2

Software Engineering models: Predictive software engineering models and its application - Model Approaches – Prerequisites - predictive and adaptive waterfall - waterfall with feedback - Sashimi - incremental waterfall - V model–System development lifecycle- Iterative vs Predictive–prototypes-Spiral- unified process Clean room - Rapid Application development principles – risk management.

Module 3

Fundamentals of Agile Development: Introduction to agility, Agile Principles, Overview of Scrum, Extreme Programming, Feature Driven development, Lean Software Development, Agile project management. Design and development practices in Agile projects, Test Driven Development, Pair Programming, Simple Design, User Stories, Agile Testing, Agile Tools - Agile design practices- Refactoring, Need and significance of Refactoring, Refactoring Techniques.

The Agile lifecycle and its impact on testing, Test-Driven Development (TDD), JUnit framework and tools for TDD, Testing user stories - acceptance tests and scenarios, Planning and managing testing cycle, Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools to support the Agile tester.

Module 4

Scrum Framework: Introduction to Scrum, Project phases, Agile Estimation, Planning game, Product backlog, Sprint backlog, Iteration planning, User story definition, Characteristics and content of user stories, Acceptance tests and Verifying stories, Project velocity, Burn down chart, Sprint planning and retrospective Daily scrum, Scrum roles - Product Owner, Scrum Master, Scrum Team, Scrum case study

Module 5

Industry Trends: Introduction to DevOps- A unified process between development and operations-Continuous Integration (CI), continuous testing, and continuous deployment - Configuration management, release management, and monitoring and learning.

Textbooks

1. Rod Stephens, "Beginning Software Engineering", Wrox Series, Wiley India Pvt Ltd, (2015)

Reference Books

1. Alistair Cockburn, "Agile Software Development: The Cooperative Game", Addison Wesley, 2nd Edition (2006).
2. Andrew Hunt, David Thomas, "The Pragmatic Programmer: From Journeyman to Master", Pearson India, 1st Edition (2008).
3. Ken Schwaber, Mike Beedle, "Agile Software Development with Scrum", Pearson (2008).
4. Lisa Crispin, Janet Gregory, "Agile Testing: A Practical Guide for Testers and Agile Teams", Addison Wesley Professional, 1st Edition (2008).
5. Mike Cohn, "User Stories Applied: For Agile Software Development", Addison Wesley, 1st Edition, (2004)
6. Pressman, R.S., "Software Engineering: A Practitioner's Approach", McGraw Hill SE, 7th Edition, (2010)

7. Robert C. Martin, "Agile Software Development, Principles, Patterns and Practices", Prentice Hall Imprint, Pearson Education, 2nd Edition(2002).
8. Rod Stephens, "Beginning Software Engineering", Wrox Series, Wiley India Pvt Ltd (2015).
9. RyPress "Ry's Git Tutorial" (Free e-book)
10. Jennifer Davis & Katherine Daniels, Effective DevOps Building a Culture of Collaboration, Affinity, and Tooling at Scale, O'Reilly Media, Inc, FirstEdition,2016
11. Gene Kim, Jez Humble, Patrick Debois, John Willis," The DevOps Handbook How to create world-class agility, reliability and security in technology organizations", IT Revolution press, First Edition,2016

Web Resources

1. https://www.edx.org/course/introduction-to-devops-transforming-and-improving
2. https://courses.edx.org/courses/course-v1:Microsoft+DEV212x+4T2017/course
3. https://guides.github.com/introduction/git-handbook/

Course Contents and Lecture Schedule

No	Topic	No. of Lecture Hours
1	**Module 1**	**9 hrs.**
1.1	What is Software Engineering - Why is software engineering important	1
1.2	Details around requirements gathering	2
1.3	Software design	1
1.4	Development, Testing	1
1.5	Deployment, Maintenance	1
1.6	Planning phase – project planning objective, software scope	1
1.7	Empirical estimation, models, COCOMO	1
1.8	Staffing and personal planning	1
2	**Module 2**	**10 hrs**
2.1	Predictive software engineering models and its application	1

2.2	Model Approaches – Prerequisites - predictive and adaptive waterfall	2
2.3	Waterfall with feedback	1
2.4	Sashimi - incremental waterfall - V model	1
2.5	Friend FunctionsSystem development life cycle - Iterative vs Predictive – prototypes	1
2.6	Spiral - unified process	2
2.7	Cleanroom - Rapid Application development principles – risk management.	2
3	**Module 3**	**8 hrs**
3.1	Introduction to agility, Agile Principles	1
3.2	Extreme Programming, Feature Driven Development, Lean Software	2
3.3	Design and development practices in Agile projects, Test Driven Development	1
3.4	Refactoring, Pair Programming, Simple Design, User Stories, Agile Testing	1
3.5	The Agile lifecycle and its impact on testing, Test-Driven Development (TDD)	2
3.6	Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools	1
4	**Module 4**	**10 hrs**
4.1	Introduction to Scrum, Project phases	1
4.2	Agile Estimation, Planning game	1
4.3	Product backlog, Sprint backlog, Iteration planning	1
4.4	User story definition, Characteristics and content of user stories, Acceptance tests	2
4.5	Project velocity, Burn down chart	1
4.6	Sprint planning and retrospective Daily scrum, Scrum roles - Product Owner	2
4.7	Scrum Master, Scrum	1
4.8	Tools for Agile project management	1

5	**Module 5**	**10 hrs**
5.1	Introduction to DevOps, A unified process between development and operations	1
5.2	Continuous Integration (CI), continuous testing	1
5.3	Continuous deployment	1
5.4	Configuration management	1
5.5	Release management	1
5.6	Monitoring and learning	1

24SJINMCA304	FUNDAMENTALS OF DATA SCIENCE	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course helps the students to learn, understand, and practice the basic concepts and techniques of data science. The students will be able to develop skills for implementing data science concepts for solving practical problems.
Prerequisite:	24SJINMCA107 Introduction to Computers & PC hardware 24SJINMCA305 Introduction to RDBMS and SQL

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the fundamental concepts of data science	K4
CO2	Understand the storage concepts and technologies for data	K2
CO3	Understand data analysis techniques for applications handling large data	K4
CO4	Understand Apache Hadoop Framework	K2
CO5	Understand processing environments and applications of big data	K4

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	2	2			3
CO2	3	2	3	3				3
CO3	3	2	3	3				3
CO4	3	2	3	3				3
CO5	3	2	3	3				3

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
100	**40**	**60**	**3 hours**

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks

Continuous Assessment Test (2 numbers) : 20 Marks

Assignment/Quiz/Course project : 12 Marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Classify the different types of data handled in bigdata processing?
2. Illustrate classification with an example

Course Outcome 2 (CO2):

1. Compare sharding and replication
2. Summarise the concepts of NoSQL databases and New SQL Databases.

Course Outcome 3(CO3):

1. Illustrate the steps involved in Data Analytics lifecycle
2. Illustrate qualitative Analysis techniques with example

Course Outcome 4 (CO4):

1. Sketch HDFS with a suitable diagram (K3)
2. Compare qualitative and quantitative analysis with suitable examples.

Course Outcome 5 (CO5):

1. Illustrate the role of data management layer in processing data
2. Illustrate the applications of bigdata

SYLLABUS

Module I

Introduction to Big Data Platform‑History of Data Management-Structuring Bigdata- Elements of Big Data, Big data Analytics.

Introduction to Data Mining: Data Mining Process, Data Mining Techniques - Clustering -

Classification - Regression - Outlier detection - Association rules - Prediction, Big Data Vs Data Mining.

Module 2

Big Data Storage Concepts: Clusters - File Systems and Distributed File Systems- Sharding – Replication – CAP Theorem – ACID – BASE

Big Data Storage Technology: On-Disk Storage Devices –Distributed File Systems- RDBMS Databases-NoSQL Databases-NewSQL Databases, In- Memory Storage Devices: In-Memory Data Grids, In-Memory Databases.

Module 3

Data Analytics Life Cycle: discovery- data preparation-model planning- model building communicate results-Operationalize.

Introduction to Big Data Analysis Techniques- Need for Analysis - Types of Analysis - Quantitative Analysis – Qualitative Analysis – Data Mining - Statistical Analysis – Machine Learning – Semantic Analysis – Visual Analysis.

Module 4

Data Analytics Tools: Introduction to Hadoop Ecosystem- Hadoop Distributed File System-HDFS Architecture-Features of HDFS-Map Reduce-Features of Map Reduce- Hadoop Yarn-HBase- Hive – Sqoop – ZooKeeper – Flume –Oozie.

Module 5

Understanding Big Data Technologies: Data Source Layer, Ingestion layer, Storage Layer, Physical Infrastructure Layer, Platform Management Layer, Security Layer, Monitoring Layer, Analytics Engine, Visualization Layer, Big Data Applications

Text Books

1. DreamTech Editorial Services, “Big Data Black Book”, Dreamtech Press, 2015Edition.
2. Thomas Erl ,Big Data Fundamentals Concepts, Drivers and Techniques”, Pearson

3. Education FirstEdition,2016
4. Data Science and Big Data Analytics, EMC Education Services, WileyPublications
5. Data Mining: Concepts and Techniques, Jiawei Han, Micheline Kamber, Morgan Kaufmann Publications.

Reference Books

1. Chandrakant Naikodi, "Managing Big Data", Vikas Publishing, 2015

MOOC

1. https://www.coursera.org/specializations/jhu-data-science
2. https://www.tutorialspoint.com/hadoop/hadoop_big_data_overview.htm

Web Resources

1. https://www.udacity.com/course/intro-to-data-science--ud359

Course Contents and Lecture Schedule

No	Topic	No. of lectures
1	**Module 1**	**8 hrs**
1.1	**Introduction to Big Data Platform** – History of Data Management- Structuring Big data **-**	2
1.2	Elements of Big Data, Big data Analytics	2
1.3	**Introduction to Data Mining:** Data Mining Process, Data Mining Techniques - Clustering - Classification – Regression	2
1.4	Outlier detection - Association rules - Prediction, Big Data Vs Data Mining	2
2	**Module 2**	**12 hrs**
2.1	**Big Data Storage Concepts:** Clusters - File Systems and Distributed File Systems- Sharding	3
2.2	Replication - CAP Theorem - ACID - BASE	3

2.3	**Big Data Storage Technology:** On-Disk Storage Devices - Distributed File Systems	3
2.4	RDBMS Databases-NoSQL Databases-NewSQL Databases In-Memory Storage Devices: In-Memory Data Grids, In-Memory Databases	3
3	**Module 3**	**10 hrs**
3.1	**Data Analytics Life Cycle:** discovery- data preparation-model planning model building- communicate results-Operationalize	3
3.2	**Introduction to Big Data Analysis Techniques-** Need for Analysis - Types of Analysis - Quantitative Analysis	3
3.3	Qualitative Analysis - Data Mining- Statistical Analysis-Machine Learning - Semantic Analysis - Visual Analysis	4
4	**Module 4**	**10 hrs**
4.1	**Data Analytics Tools:** Introduction to Hadoop Ecosystem - Hadoop Distributed File System-HDFS Architecture	3
4.2	**Features of HDFS** - Map Reduce- Features of Map Reduce-Hadoop Yarn	4
4.3	HBase- Hive - Sqoop - ZooKeeper - Flume - Oozie.	3
5	**Module 5**	**8 hrs**
5.1	**Understanding Big Data Technologies:** Data Source Layer, Ingestion layer, Storage Layer, Physical Infrastructure Layer	4
5.2	Platform Management Layer, Security Layer, Monitoring Layer, Analytics Engine, Visualization Layer, Big Data Applications	4

24SJINMCA306	VISUAL PROGRAMMING	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course provides a solid theoretical foundation that furnishes the student about the visual programming concepts using .NET library, and to enable them to design and code visual programs
Prerequisite:	User Interface Design

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the fundamentals and architecture of visual programming environments	K5
CO2	Design and develop GUI-based applications using event-driven programming concepts.	K5
CO3	Understand the principles of object-oriented programming in the context of visual environments	K5
CO4	Implement database connectivity and data handling in visual applications.	K4
CO5	Develop, test, and deploy complete visual applications using modern frameworks/tools.	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	2	–	–	–	2
CO2	2	3	3	3	2	1	–	2
CO3	3	2	2	3	1	–	–	2
CO4	2	3	3	3	2	1	–	2
CO5	2	2	3	3	3	2	1	3

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks

Continuous Assessment Test (2 numbers) : 20 Marks

Assignment/Quiz/Course project : 12 Marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module in which students should answer anyone. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe the concept of Event-Driven Programming with examples in Visual Studio
2. Explain the architecture of Visual Studio IDE and describe its main components.
3. Compare event-driven and procedural paradigms with an example of a simple form-based application. Discuss how this transition improves user interactivity.

Course Outcome 2 (CO2)

1. Design a form-based application with text boxes, labels, and buttons to perform basic arithmetic operations using event handling
2. Develop a small GUI application (e.g., Student Entry Form) with form validation, multiple forms, and message boxes. Describe the event flow.
3. Create a GUI that responds to mouse and keyboard events

Course Outcome 3 (CO3):

1. Illustrate how inheritance can be used to create reusable GUI components in a visual programming tool
2. Design a simple GUI application that demonstrates the concept of class and object interaction (e.g., a calculator or student record form).
3. Write an example to show how polymorphism is implemented in a form-based application.

Course Outcome 4 (CO4):

1. Write the steps to connect a GUI application to a relational database using a suitable database connector.
2. Define CRUD operations. How are they implemented in visual applications.
3. Demonstrate how data from a database can be displayed using data-bound controls (e.g., DataGridView, ListView).

Course Outcome 5 (CO5):

1. Explain the process of debugging and testing GUI applications in Visual Studio
2. Define application deployment in the context of Visual Studio.
3. Demonstrate how to build and test a C# Windows Forms application in Visual Studio.

Syllabus

Module 1 – Introduction to Visual Programming

Evolution of programming paradigms – From procedural to visual programming, features of visual programming environments, Event-driven programming concepts, Overview of commonly used visual programming tools, Structure of a GUI-based application, designing user interfaces using forms, controls, and properties.

Module 2 – Event Handling and Controls

Event-driven architecture and message loops, Handling user interactions (mouse, keyboard, form events), Standard controls – Labels, Textboxes, Buttons, ComboBoxes, ListBoxes, Menus, Toolbars, Status Bars, Timer control and validation techniques, working with dialogs and message boxes, Managing multiple forms and navigation.

Module 3 – Object-Oriented Programming in Visual Environment

Classes, objects, and methods in GUI applications, Encapsulation and inheritance in visual programming, Exception handling in GUI applications, Collections and generics, File handling – reading and writing files, Modularization and code reuse (user-defined controls and components)

Module 4 – Database Connectivity and Data Access

Connecting to databases using ADO.NET / JDBC / similar APIs, Executing SQL commands from visual applications, Data binding and managing data grids, Performing CRUD operations (Create, Read, Update, Delete), Error handling and transactions, Case study: Mini database application

Module 5 – Advanced Topics and Application Development

Integration of multimedia components (images, audio, video), Using external libraries and APIs, Introduction to web-enabled and cross-platform visual applications Debugging and testing GUI applications, Packaging and deployment of desktop applications, Mini Project: Developing a GUI-based application integrating UI, logic, and database

Reference Books

1. Microsoft Visual C# Step By Step, 10th Edition: Author: John Paul Mueller and Bill Wagner (2022).
2. The Complete Visual C# Programmer's Guide: Author: Philip Järvinen (2002)
3. Database Programming with C# (The Expert's Voice) by Carsten Thomsen.

MOOC Courses

1. Introduction to Programming With C# (Coursera) https://www.coursera.org/learn/introduction-to-programming-with-c-sharp.
2. Microsoft – Learn C# resource: https://dotnet.microsoft.com/en-us/learn/csharp

Web Resources

1. https://learn.microsoft.com/en-us/dotnet/csharp/
2. https://www.dotnetperls.com/
3. https://www.geeksforgeeks.org/csharp-programming-language/

Course Contents and Lecture Schedule

	Topic	**No. of lectures**
1	**Module 1**	**10 hrs.**
1.1	Evolution of programming paradigms – From procedural to visual programming	2
1.2	Features of visual programming environments	1
1.3	Event-driven programming concepts	1
1.4	Overview of commonly used visual programming tools	2
1.5	Structure of a GUI-based application	2
1.6	Designing user interfaces using forms, controls, and properties	2
2	**Module 2**	**8 hrs.**

2.1	Event-driven architecture and message loops	2
2.2	Handling user interactions (mouse, keyboard, form events)	2
2.3	Standard controls – Labels, Textboxes, Buttons, ComboBoxes, ListBoxesMenus, Toolbars, Status Bars,Timer control and validation techniques	2
2.4	Working with dialogs and message boxes,Managing multiple forms and navigation	2
3	**Module 3**	**10 hrs.**
3.1	Classes, objects, and methods in GUI applications	2
3.2	Encapsulation and inheritance in visual programming	2
3.3	Exception handling in GUI applications	2
3.4	Collections and generics,File handling – reading and writing files	2
3.5	Modularization and code reuse (user-defined controls and components)	2
4	**Module 4**	**11 hrs.**
4.1	Connecting to databases using ADO.NET / JDBC / similar APIs	2
4.2	Executing SQL commands from visual applications	1
4.3	Data binding and managing data grids	2
4.4	Performing CRUD operations (Create, Read, Update, Delete)	2
4.5	Error handling and transactions	1
4.6	Case study: Mini database application	3
5	**Module 5**	**9 hrs.**
5.1	Integration of multimedia components (images, audio, video),Using external libraries and APIs	2
5.2	Introduction to web-enabled and cross-platform visual applications Debugging and testing GUI applications	2
5.3	Packaging and deployment of desktop applications	1
5.4	Mini Project: Developing a GUI-based application integrating UI, logic, and database	4

24SJINMCA308	DESIGN AND ANALYSIS OF ALGORITHMS	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	The syllabus is prepared with the view of equipping the MCA Graduates with the techniques for effective problem solving in computing. The course content is decided to cover the fundamentals of Design and Analysis of Algorithm which can be taught within the given slots in the curriculum.
Prerequisite:	24SJINMCA105-Introduction to Programming, 24SJINMCA108-Problem Solving and Structured Programming, 24SJINMCA209-Data Structures

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Implement design principles and analyze the asymptotic performance of algorithms.	K3
CO2	Derive and solve recurrences describing the performance of divide-and-conquer algorithms and greedy algorithms	K4
CO3	Synthesize dynamic-programming algorithms, and analyze them	K4
CO4	Explore various algorithm design instances like back tracking, Branch & Bound in solving common analytical problems.	K5
CO5	Infer basic computational concepts and the complexity classes P, NP, and NP-Complete.	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	1					1
CO2	3	3	2					1
CO3	3	3	2					1
CO4	3	3	2					1
CO5	3	3	2					1

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test(2numbers) : 20 marks Assignment/Quiz/Course
project : 12 marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Compare Priori and Posteriori analysis of algorithms
2. Illustrate the Properties of Algorithms.
3. State master's theorem. Formulate T (n) = 4T (n/2) + n2 using Master's theorem.

Course Outcome 2 (CO2):

1. Describe the Control Abstraction for Divide and Conquer
2. Illustrate merge sort algorithm with proper code sequence
3. Illustrate Binary Search Algorithm
4. Derive an element from the given array using Divide and Conquer Strategy

Course Outcome 3(CO3):

1. Compare Divide and Conquer Method and Dynamic Programming
2. Compose an algorithm to find the shortest path in a weighted graph
3. Outline Principle of Optimality

Course Outcome 4 (CO4):

1. Define the Sum of Subsets Problem with suitable example
2. Illustrate the control abstraction of Backtracking
3. Summarize Least Cost (LC) Search with suitable example

Course Outcome 5 (CO5):

1. Explain non-deterministic algorithm
2. Compare Tractable and Intractable Problems.
3. With the help of suitable code sequence convince Clique problem is an example of NP- Complete Problem

Syllabus

Module 1-Introduction to Algorithm Analysis:

Definition of Algorithms, Properties of Algorithms, Types of Algorithm's Analysis- Priori and Posteriori analysis of algorithms, Time and Space Complexity- Elementary Operation and Complexity Estimation of Simple Algorithms -Asymptotic notations - Recurrence Relations - Solution of Recurrence Relations – Iterative Method - Recurrence Tree Method - Master Theorem (Proof not required)

Module 2-Divide and Conquer Method:

Control Abstraction for Divide and Conquer, Binary Search, Merge Sort, Quick sort, Finding Maximum and minimum Greedy Strategy: - Control Abstraction for Greedy Strategy, The Knapsack Problem, Minimum cost spanning Trees- Prims' and Kruskal's Algorithms, Job Sequencing Problem

Module 3-Dynamic Programming:

Characteristics of Dynamic Programming, Divide and Conquer Vs Dynamic Programming, The Principle of Optimality- All Pair Shortest Path Problem-Travelling Sales Person Problem,

Module 4- Backtracking:

Control Abstraction for Backtracking - The N Queen's Problem, Sum of Subsets Problem.

Branch and Bound: Least Cost (LC) Search - Control Abstraction ,The 15 Puzzle Problem

Module 5- Introduction to Computational Complexity:

Tractable and Intractable Problems. NP-Hard and NP -complete problems - Basic concepts, non-deterministic algorithm, class of NP- hard and NP-complete. Study of NP complete problems-Knapsack Problem- Clique Problem, Vertex Cover Problem.

Text Books

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Orient Longman, Universities Press, 2nd Edition(2008)

2. Rajesh K.Shukla, "Analysis and Design of Algorithms, A Beginner's Approach", Wiley (2015)

3. Dr. R Vijayakumar, Dr.Juby Mathew, "Algorithm Design, Foundation, Analysis and Examples", Vimala Books and Publications,Ist Edition(2016)

Web Resources

1. https://www.javatpoint.com/daa-tutorial

2.https://www.tutorialspoint.com/design_and_analysis_of_algorithms/index.htm

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**8 hrs.**
1.1	Introduction, Definition of Algorithms, Properties of Algorithms , Types of Algorithm's Analysis- Priori and Posteriori analysis of algorithms	1
1.2	Space Complexity	1
1.3	Time Complexity and Complexity Estimation of Simple Algorithms	2
1.4	Asymptotic notations	1
1.5	Solution of Recurrence Relations - Iterative Method	1
1.6	Solution of Recurrence Relations - Recurrence Tree Method	1
1.7	Solution of Recurrence Relations -Master Theorem	1
2	**Module 2**	**15 hrs.**
2.1	**Divide and Conquer Method:** Introduction, Control Abstraction for Divide and Conquer	1
2.2	Binary Search	1
2.3	Merge sort	2
2.4	Quick sort	2
2.5	Finding Maximum and minimum	2

2.6	Control Abstraction for Greedy Strategy,The Knapsack Problem	1
2.7	Prims' Algorithm	2
2.8	Kruskal's Algorithm	2
2.9	Job Sequencing Problem	2
3	**Module 3**	**7 hrs.**
3.1	Introduction to Dynamic Programming,Divide and Conquer Vs Dynamic Programming.	2
3.2	The Principle of Optimality	1
3.3	All Pair Shortest Path Problem	2
3.4	Travelling Sales Person Problem,	2
4	**Module 4**	**10 hrs.**
4.1	Control Abstraction for Backtracking	1
4.2	The NQueen's Problem	2
4.3	Sum of Subsets Problem.	2
4.4	Least Cost (LC) Search - Control Abstraction ,	3
4.5	The 15 Puzzle Problem	2
5	**Module 5**	**8 hrs.**
5.1	Introduction to Computational Complexity: Tractable and Intractable Problems.	1
5.2	NP-Hard and NP -complete problems - Basic concepts, non-deterministic algorithm	1
5.3	class of NP- hard and NP- complete.	2
5.4	Study of NP complete problems - Knapsack Problem	1
5.5	Study of NP complete problems -Clique Problem	1
5.6	Study of NP complete problems - Vertex Cover Problem.	2

24SJINMCA310	MINI PROJECT 1	CATEGORY	L	T	P	CREDIT
		PROJECT	0	0	4	2

Preamble:	The mini project is designed to develop practical ability and knowledge about tools/techniques in order to solve the actual problems related to the industry, academic institutions or similar areas.
Prerequisite:	Knowledge in software engineering principles and programming skills

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Practice acquired knowledge within the chosen area of technology for project	K3
CO2	Identify, discuss and justify the technical aspects of the chosen project with a comprehensive and systematic approach.	K4
CO3	Reproduce, improve and refine technical aspects for engineering projects.	K4
CO4	Work as an individual or in a team in development of technical projects.	K3
CO5	Communicate and report effectively project related activities and findings	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	2	1	1	1	2
CO2	2	3	3	2	1	2	1	2
CO3	2	2	3	3	1	2	1	2
CO4	1	1	2	1	3	2	1	2
CO5	1	2	2	1	3	2	1	2

Syllabus

Students can take up any application level/system level experimental design/implementation tasks of relatively minor intensity and scope as compared to the major-project, pertaining to a relevant domain of study. Projects can be chosen either from the list provided by the faculty or in the field of interest of the student. At the end of each phase, presentation and demonstration of the project should be conducted, which will be evaluated by a panel of examiners. A detailed project report duly approved by the guide and Head of the department in the prescribed format should be submitted by the student for final evaluation. Students as individual or a team of maximum 4 members may be trusted with providing solution to similar problems

Mini project will give an opportunity for the students to prepare for the main project and to achieve some of the objectives of the main project. This can also be used as an opportunity for producing and distributing socially useful software's.

Internal Continuous Assessment (100 marks)

10% (10) - Project Synopsis / Proposal Evaluation

20% (20) - 1st Interim Project Evaluation

20% (20) - 2nd Interim Project Evaluation

50% (50) - End Semester Internal Project work & Project Report Evaluation by the Panel Constituted by the Head of the department.

General Guidelines:

● Students shall identify Real-Life Projects which are relevant and useful to the society or industry.
● The project shall be an individual/ Group (Maximum of 4 members) project and must be done in-house. The student has to spend time in the lab for the project work. Attendance as per MCA regulations is applicable for submitting the project for final evaluation.
● However, in exceptional cases, students may be permitted to undertake the project work at industry premises, if offered by the organization. The company proposal should be approved by the Department Internship Scrutiny Committee, constituted by the Head of the Department. This internship shall be completed during the semester break period between Semester 5 and Semester 6.

- In such cases, the student is required to produce a letter from the organization before starting the project and an internship scrutiny committee constituted by the head of the department shall make the decision on permission. Industries and training institutes that offer project work for a fee shall not be permitted.

● Students shall submit project synopsis and get prior approval from the Project (Faculty) Supervisor before the project work begins.
● If there is a customer for the project then he/she will be the Product Owner (External Guide) and a faculty from the department will be the Internal Guide. If there is no such customer then the Internal Guide himself/herself shall act as the Product Owner.
● A faculty / technical staff shall act as the Scrum Master to continuously monitor the project development. Periodic meetings, of less than 15 minutes, at the convenience of the Scrum Master are to be highly encouraged. Ensure such meetings occur once in three days.
● The student shall maintain a Scrum Book (Rough Record) which has to be divided into 4 parts – (i) Product Backlog (ii) Database & UI Design (iii) Testing & Validation and (iv) Details of Versions. Make dated entries in the corresponding part at regular intervals. The corrections and comments from Product Owner and Scrum Master should be clearly indicated with the Date.
● Test Driven Development methodology may be practiced for the project development. BugZilla, BackLog or any such tool may be used for Bug Tracking.
● Git shall be used for Version Control and Git commit history may be verified as part of project evaluation.
● LaTeX or an equivalent tool shall be used for preparing Presentations and Project Report.
● Students shall be encouraged to publish their work in journals and due credit shall be given to such students.

● For the externally done projects, periodic confidential progress report and attendance statement shall be collected from the External Guide and be reviewed by the Project Supervisor.
● Set a sprint as two weeks, ensure biweekly reviews. A review shall not exceed 30 minutes. A demo to the Product Owner (Project Guide) is mandatory in every review.
● Interim evaluations of the project's progress shall be conducted by a Project Assessment Board as part of internal assessment. Two such evaluations are desirable. Scrum reviews shall not be sacrificed for such presentations.
● The Project Assessment Board shall be constituted by the Head of the Department with the following three members.

Chairman:
1. Head of the Department
Members:
2. Project supervisor/s of the student
3. One Senior faculty member from the Department

● At the end of the semester, Project Assessment Board evaluation shall be there on the entire project development activities.

Week	Schedule
(May be scheduled inline with the academic calendar)	
1	Selection of Topic, Submission of project synopsis and getting approval Meeting of Development Team including Scrum Master with Product Owner (Project Guide)
2	Commencement of the Project. 4
3	First Sprint release and Scrum Review by the Product Owner (Project Guide)
4	Second Sprint release and Scrum Review by the Project Guide First interim evaluation by the Project Assessment Board
5	Third Sprint release and Scrum Review by the Project Guide
6	Fourth Sprint release and Scrum Review by the Project Guide
7	Second interim evaluation by the Project Assessment Board
8	Fifth Sprint release and Scrum Review by the Project Guide
9	Submission of project report, with Scrum Book Final project presentation Evaluation by the Project Assessment Board

24SJINMCA332	OPEN SOURCE PLATFORMS - LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	1

Preamble:	This course aims to give a basic understanding of open source platforms to the students, along with various technologies in practice.
Prerequisite:	

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Have a good understanding of Open Source Community and Develop open-source Software Projects using Python	K6
CO2	Participate in developing Open-Source Projects using Python	K6
CO3	Have an understanding of intellectual property rights, open-source licensing, and the implications of using open source tools in developing web-enabled software like Word Press and Drupal	K6
CO4	Develop Shell Programs for system administration	K6
CO5	Have an understanding of open source software tools GitHub	K6

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	1	2	2	1	-	1	2
CO2	1	3	2	2	3	1	-	2
CO3	1	1	2	1	1	2	3	2
CO4	3	1	2	3	-	-	-	1
CO5	1	1	2	3	2	1	-	2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks

Continuous Assessment Test (2 numbers) : 20 Marks

Assignment/Quiz/Course project : 12 Marks

End Semester Examination Pattern:

Lab exam will be conducted by an external examiner.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Develop a python program to illustrate the use of variables, data types, operators and expressions
2. Develop a python program to illustrate the use of Control Structures
3. Develop a python program to demonstrate String manipulation

Course Outcome 2 (CO2):

1. Design, Develop and implement a menu driven program in python to illustrate the Functions.
2. Design, Develop and implement program to illustrate the use of built in input- output functions in python.
3. Design, Develop and implement a menu driven program in python to illustrate file and its operations.
4. Design and develop python program to Passing Information using GET method.
5. Design and develop python program to Passing Information using POST method.

Course Outcome 3(CO3):

1. Design, Develop and implement a website in Drupal/ WordPress and Customize it.

Course Outcome 4 (CO4):

1. Develop Shell Programs for system administration

Course Outcome 5 (CO5):

1. Illustrate the use of Git by pushing the changes in local repository to remote repository
2. Illustrate the use of Git by pulling the changes in local repository to remote repository
3. Illustrate the use of Git by merging branches
4. Use GIT and gain knowledge in using version control

Syllabus

Python Variables, Strings, Data types – List, tuple, Dictionary, set, Operators, Expressions, Control Structures

Python Functions, Python Input–Output, Python Files, CGI Programming- Passing Information using GET method, CGI Programming-Passing Information using POST method, GUI Programming Basics –Tkniter Label, Message, Text and Button Designing a Site, customizing a Site, Content Management, Adding Features to Site – Wordpress/ Drupal Administration Level

Introduction to Shell scripting – Experiment with shell scripts mainly for administrative tasks like user creation in bulk, changing file permissions recursively, creating files in bulk, deleting folders and sub folders etc...

1. Commands
1. echo,read
2. more,less
3. man
4. chmod,chown
5. cd, mkdir, pwd, ls,find
6. cat, mv, cp,rm
7. wc, cut,paste
8. head, tail, grep,expr
9. Redirections &Piping
10. useradd, usermod, userdel,passwd
11. tar

Reference Books

1. Dr. Jeeva Jose, “Taming Python by Programming”, Khanna Publishing (2017)
2. Jon Loeliger, “Version Control with Git”, O’Reilly mediaInc.(2009)
3. Matt Glaman, “Drupal 8 Cookbook”, PackTPublishing (2016)
4. B. M. Harwani, Unix and Shell programming”, Oxford UniversityPress(2013)

Web Resources

1. https://www.hostinger.in/tutorials/wordpress/
2. https://www.hostinger.in/tutorials/drupal
3. https://docs.python.org/2/tutorial/
4. https://documentation.libreoffice.org/en/english-documentation/
5. https://docs.github.com/en

MOOC

1. https://www.coursera.org/learn/interactive-python-1#pricing
2. https://developers.google.com/edu/python/
3. https://www.edx.org/course/introduction-to-computer-science-and-programming-using- python
4. http://www.learnpython.org/
5. https://www.wp101.com/course/wordpress101/

List of Lab Experiments

1. Design, Develop and implement python program to illustrate the use of variables, data types, operators and expressions
2. Write a python function moderate Days that is given a dictionary containing daily temperature for each day of the week, and returns the list of the days in which the average temperature was between 70 &79degrees.
3. Design, Develop and implement python program to illustrate the use of Control Structures
4. Design, Develop and implement python program to demonstrate String manipulation
5. Design, Develop and implement a menu driven program in python to illustrate the Functions
6. Design, Develop and implement program to illustrate the use of built in input- output functions in python
7. Design, Develop and implement a menu driven program in python to illustrate file and its operations
8. Design, Develop and implement a website in Drupal/ WordPress and Customize it.
9. Develop Shell Programs for system administration
10. Illustrate the use of Git by pushing the changes in local repository to remote repository
11. Illustrate the use of Git by pulling the changes in local repository to remote repository
12. Illustrate the use of Git by merging branches

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**7 hrs.**
1.1	Python Variables	1
1.2	Strings	1
1.3	Data types – List, tuple, Dictionary, set	2
1.4	Operators	1
1.5	Expressions	1
1.6	Control Structures	1
2	**Module 2**	**10 hrs.**
2.1	Python Functions	2
2.2	Python Input – Output	1
2.3	Python Files	2
2.4	CGI Programming - Passing Information using GET method	2
2.5	CGI Programming - Passing Information using POST method	1
2.6	GUI Programming Basics –Tkniter Label, Message, Text and Button	2
3	**Module 3**	**8 hrs.**
3.1	Designing a Site	2
3.2	Customizing a Site	2
3.3	Content Management	2
3.4	Adding Features to Site	2
4	**Module 4**	**12 hrs.**

4.1	Experiment with shell scripts mainly for administrative tasks like user creation in	2
4.2	Experiment with shell scripts mainly for administrative tasks like changing file	2
4.3	Experiment with shell scripts mainly for administrative tasks like creating files in	2
4.4	Experiment with shell scripts mainly for administrative tasks like deleting folders	2
4.5	Experiment with shell scripts for Remote access	2
4.6	Experiment with shell scripts for Scheduling Using cron and at	2
5	**Module 5**	**11 hrs.**
5.1	Using Git for version Control	2
5.2	creating a Local repository, adding files to the Staging area, committing changes	1
5.3	creating a Remote repository, connecting Local repository with Remote	2
5.4	pushing the changes to the Remote repository	1
5.5	pulling the changes from the Remote repository	1
5.6	organizing your Repository with Branches	2
5.7	Working in a team – Branches And merging.	2

24SJINMCA334	VISUAL PROGRAMMING LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	0	4	1

Preamble:	The **Visual Programming Lab** aims to provide hands-on experience in developing interactive, event-driven applications using **C#**. This lab enables students to design and implement graphical user interfaces (GUIs) using Windows Forms and explore the principles of **event handling, object-oriented programming,** and **database connectivity.** Through practical exercises and mini projects, students will learn to integrate user interface design with backend logic and data management, ultimately developing complete, deployable desktop applications.
Prerequisite:	Introduction to Programming, User Interface Design

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Design and develop basic GUI-based applications using forms, controls, and properties to understand the fundamentals of visual programming.	K5
CO2	Implement event-driven programming in C# to handle user interactions using standard controls, timers, and multiple forms.	K6
CO3	Apply object-oriented concepts such as classes, inheritance, exception handling, and file operations in C# visual applications.	K6
CO4	Integrate database connectivity using databases like ADO.NET or similar technologies to perform CRUD operations and manage data within GUI applications	K6
CO5	Develop and deploy complete C# desktop applications integrating UI design, business logic, multimedia components, and database connectivity.	K6

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	2	1	-	-	1
CO2	3	2	3	2	1	-	-	1
CO3	3	2	3	3	2	1	-	2
CO4	2	2	3	3	2	2	-	2
CO5	2	2	3	3	3	2	1	2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 Marks
Continuous Assessment Test (2 numbers)	: 20 Marks
Assignment/Quiz/Course project	: 12 Marks

End Semester Examination Pattern

Lab exam will be conducted by an internal examiner.

Course Level Assessment

Course Outcome 1 (CO1):

1. Design and develop Windows Forms applications in C# to familiarize form design, controls, layout management, and property settings.

Course Outcome 2 (CO2):

1. Design, develop, and implement event-driven programs in C# to handle user interactions through mouse, keyboard, and control events.
2. Develop applications using standard controls such as labels, buttons, combo boxes, list boxes, menus, and timers with validation techniques.

Course Outcome 3 (CO3):

1. Design and implement programs applying object-oriented principles such as classes, inheritance, and encapsulation within GUI-based C# applications.
2. Develop C# applications that handle runtime exceptions and perform file operations for data storage and retrieval.

Course Outcome 4 (CO4):

1. Design and implement GUI-based applications that connect to databases using suitable technologies.
2. Execute database operations such as Create, Read, Update, and Delete (CRUD) and display data using appropriate controls.

Course Outcome 5 (CO5):

1. Develop a complete GUI-based C# application integrating user interface, business logic, multimedia, and database components. (K6)
2. Debug, test, and package the developed desktop application for deployment. (K6)

Syllabus

Event-driven programming concepts, introduction to visual programming environments, designing user interfaces using forms, controls, and properties. Event handling and user interaction through mouse, keyboard, and control events. Implementation of object-oriented programming concepts such as classes, inheritance, exception handling, and file operations in GUI applications. Integration of database connectivity using databases like ADO.NET or similar technologies to perform CRUD operations and manage data. Development of complete GUI-based applications incorporating UI design, logic, multimedia, and deployment techniques.

Reference Books

1. Komalavalli C., Sanjib K. Sahu, *Essentials of .NET Programming: Theory and Application*, Ane Books Pvt. Ltd., 2016.
2. Shirish Chavan, *Visual Basic .NET*, First Edition, Pearson Education, 2004.
3. Steven Holzner, *Visual Basic .NET Programming Black Book*, New Edition, Dreamtech Press, 2005.
4. Andrew Troelsen and Philip Japikse, *Pro C# 10 with .NET 6: Foundational Principles and Practices in Programming*, Apress, 2022.
5. Joe Mayo, *C# Programming: A Step-by-Step Guide for Beginners*, Microsoft Press, 2020.

Web Resources

1. https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/ — Microsoft's interactive "Tour of C#" tutorials for beginners.
2. https://www.w3schools.com/cs/index.php — W3Schools C# tutorial covering variables, data types, control statements, etc.
3. https://www.simplilearn.com/tutorials/c-sharp-tutorial/c-sharp-gui — Simplilearn tutorial focused on C# GUI programming (forms, controls, events) in desktop apps.
4. https://code.visualstudio.com/docs/csharp/get-started — Getting started guide for C# development in VS Code (setup, basic project) which can help students set up their environment.
5. https://www.codementor.io/collections/learn-c-sharp-online-bwbavow0y — A curated list of online tutorials and resources for learning C#.

List of Lab Experiments

1. Create a simple Windows Form with labels, text boxes, and buttons to display “Hello World”.
2. Design a form to input and display student details using various controls.
3. Develop an application to perform arithmetic operations (Addition, Subtraction, Multiplication, Division).
4. Create a GUI-based temperature converter (Celsius ↔ Fahrenheit).
5. Design an application to calculate simple and compound interest.
6. Create a form to compute area and perimeter of different shapes (circle, rectangle, triangle).
7. Implement event handling for mouse and keyboard actions (MouseMove, KeyPress).
8. Develop a login form that validates username and password with message boxes.
9. Create an application using RadioButtons and CheckBoxes to display user-selected options.
10. Design a simple quiz application with timer control and score display.
11. Implement a calculator using menu and toolbar controls.
12. Develop an application with multiple forms and navigation buttons (Next/Previous).
13. Create a class for “Employee” and display employee details using objects.
14. Implement inheritance with a base class “Person” and derived classes “Student” and “Teacher”.
15. Develop an application to demonstrate exception handling during user input.
16. Create a program to read and write data to a text file using file dialog boxes.
17. Design a GUI that stores user data to a file and retrieves it when reopened.
18. Design a GUI application to connect to a database and retrieve records.
19. Implement CRUD (Create, Read, Update, Delete) operations using a connected or disconnected database model.
20. Create a student management system with data grid display and data validation.
21. Develop an application to search records dynamically using user input.
22. Handle database exceptions and transactions in GUI-based operations.
23. Create an application that plays audio or displays an image in a form.

24. Develop a notepad-like text editor application with menu options (Open, Save, Clear, Exit).
25. Mini Project: Design and implement a complete GUI-based C# application integrating UI design, logic, and database (e.g., Library Management, Attendance Tracker, or Expense Manager).

Course Contents and Lecture Schedule

No.	Topic	No. of Hours
1	**Cycle 1**	**4 hrs.**
1.1	Evolution of programming paradigms – From procedural to visual programming	2
1.2	Features of visual programming environments, structure of GUI-based applications, and designing user interfaces using forms, controls, and properties	2
2	**Cycle 2**	**8 hrs.**
2.1	Event-driven architecture and message loops	2
2.2	Handling user interactions (mouse, keyboard, and control events)	3
2.3	Standard controls – Labels, TextBoxes, Buttons, ComboBoxes, ListBoxes, Menus, Toolbars, Status Bars, Timer control, and validation techniques	3
3	**Cycle 3**	**12 hrs.**
3.1	Classes, objects, methods, and encapsulation in GUI applications	4
3.2	Inheritance, exception handling, and collections in C# GUI applications	4
3.3	File handling, modularization, and code reuse (user-defined controls and components)	4
4	**Cycle 4**	**12 hrs.**
4.1	Integrating database connectivity using databases like ADO.NET or similar technologies	5
4.2	Executing SQL commands, performing CRUD operations, data binding, and handling transactions	7
5	**Cycle 5**	**12 hrs.**
5.1	Integration of multimedia components (images, audio, video)	3
5.2	Using external libraries and APIs, debugging, and testing GUI applications	4
5.3	Packaging and deployment of desktop applications; Mini project integrating UI, logic, and database	5

ST.JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

MCA Integrated

SEMESTER 7

		CATEGORY	L	T	P	CREDIT
24SJINMCA401	**M-COMMERCE**	**CORE**	**3**	**1**	**0**	**4**

Preamble:	The purpose of this course is to make students familiar with various aspects of Mobile Commerce for performing business transactions through mobile handheld devices.
Prerequisite:	Nil

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Ability to understand and explain Mobile Commerce concepts.	K3
CO2	Understand the Mobile Commerce Technology	K3
CO3	Familiarize the Key Players and products of Mobile Commerce	K4
CO4	Access the mobile payment systems.	K4
CO5	Discuss the security and privacy issues in Mobile Commerce.	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1					2
CO2	3	3	2	2				2
CO3	2	2	2	1	1	1		2
CO4	2	3	3	3		2	2	2
CO5	2	2	2	3			3	2

Continuous Internal Evaluation Pattern:

Attendance: 8 marks

Continuous Assessment Test (2 numbers): 20 marks

Assignment/Quiz/Course project: 12 marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe Wireless Communication Technology
2. Explain Mobile portals
3. Summarize Mobile Commerce Business Models

Course Outcome 2 (CO2):

4. Differentiate 4G and 5G systems
5. Illustrate Mobile Communication Standards
6. Outline Communication Systems

Course Outcome 3 (CO3):

7. Show different types of Mobile Devices
8. Evaluate challenges of Mobile Banking
9. Design Mobile Banking Business Models.

Course Outcome 4 (CO4):

10. Outline privacy and security issues of Mobile Tickets
11. Analyse challenges of Mobile Computing
12. Summarize different types of Mobile Payments

Course Outcome 5 (CO5):

13. Explain Fraud Detection in M Commerce
14. Outline Mobile Security Mechanism
15. Describe legal aspects of mobile devices

SYLLABUS

Module 1

Introduction to Mobile Commerce- Wireless Communication Technology, Scope, Principles of Mobile Commerce, Benefits, Limitations, Mobile Commerce Framework, Mobile Commerce Business Models, Mobile Banking and Payments, E-Commerce vs M-Commerce, Mobile commerce services - Types, Mobile portals, Mobile Commerce applications - Applications of Mobile Commerce in industry, Mobile Application Development.

Module 2

Mobile Commerce Technology: Wireless and Mobile Communication - Communication

Systems, Wireless Communication, Satellite Communication, Mobile Communication Systems. Digital Cellular Technology - Cellular Networks, Mobile Phone Cellular Network, Mobile Access Technology - Mobile Communication Standards-Evolution, 4G and 5G systems - 4G Systems, IMT Advanced Standards, 4G features, 4G technologies, LTE - Advanced, 4G Objectives and Goals, 5G systems.

Module 3

Mobile Devices and Banking: Types of Mobile Devices, Mobile Service Providers: Mobile Network Operators, Mobile Commerce Service Providers. Mobile Banking: Mobile Banking Business Models, Mobile Banking Technologies, Mobile Banking Services, Advantages of Mobile Banking, Challenges of Mobile Banking, Applications.

Module 4

Tickets on Mobile and Mobile Payment: Mobile Ticketing, Applications of Mobile Tickets, Advantages of Mobile Tickets, Privacy and Security Issues, Mobile Ticketing Apps, Mobile Ticket Providers, Mobile Payment: Characteristics of Mobile Payment Systems, Mobile Payment Models, Types of Mobile Payments, Security Issues, Mobile Payment Service Providers.

Mobile Computing: Nomadic or Ubiquitous Computing, Applications of Mobile Computing, Challenges of Mobile Computing, Mobile Computing Software Platforms, Business Applications of Mobile Computing.

Module 5

Security and Privacy Issues, Legal Aspects: Mobile Security Concepts, Mobile Security Mechanism, Mobile Network Security, Mobile Information Security, Mobile Device Security, Legal Aspects: Mobile Device Related Laws.

Future of Mobile Commerce: Future of Mobile Commerce, Mobile Commerce and Consumer Acceptance, Growth of Mobile Value-Added Services, Mobile Fraud Detection, Future Issues.

Text Books

1. Karabi Bandyopadhyay, Mobile Commerce :2013 - PHI Learning Pvt. Ltd.

Reference Books

1. E. Brian Mennecke, J. Troy Strader, "Mobile Commerce: Technology, Theory and Applications", Idea Group Inc., IRM press, 2003.

2. David Taniar, "Mobile Computing: Concepts, Methodologies, Tools, and Applications" Information Science Reference,2009.

MOOC

Web Resources

1. https://www.tutorialspoint.com/mobile_marketing/m_commerce.htm
2. https://www.simicart.com/blog/mobile-commerce-guide/

Course Contents and Lecture Schedule

Topic	No. of lectures
Module 1	**9 hrs**
Introduction, Wireless Communication Technology, Scope, Principles of Mobile Commerce, Benefits, Limitations, Mobile Commerce Framework.	2
Mobile Commerce Business Models	1
Mobile Banking and Payments, E-Commerce vs M- Commerce	2
Mobile Commerce Services - Types, Mobile portals	2
Mobile Commerce Applications - Applications of Mobile Commerce in industry, Mobile Application Development	2
Module 2	**9 hrs**
Mobile Commerce Technology,Wireless and Mobile Communication - Communication Systems, Wireless Communication, Satellite Communication, Mobile Communication Systems.	2
Digital Cellular Technology - Cellular Networks, Mobile Phone Cellular Network	2
Mobile Access Technology - Mobile Communication Standards-Evolution, 4G and 5G systems -4G Systems, IMT Advanced Standards	2
4G features, 4G technologies, LTE -Advanced, 4G Objectives and Goals, 5G systems	3
Module 3	**6 hrs**
Mobile Device: Types of Mobile Devices, Mobile Service Providers: Mobile Network Operators	2
Mobile Commerce Service Providers. Mobile Banking: Mobile Banking Business Models	1
Mobile Banking Technologies, Mobile Banking Services, Advantages of Mobile Banking	2

Challenges of Mobile Banking, Applications.	1
Module 4	**12 hrs**
Tickets on Mobile:Mobile Ticketing, Applications of Mobile Tickets, Advantages of Mobile Tickets.	1
Privacy and Security Issues, Mobile Ticketing Apps, Mobile Ticket Providers	2
Mobile Payment: Characteristics of Mobile Payment Systems, Mobile Payment Models	2
Types of Mobile Payments, Security Issues, Mobile Payment Service Providers.	2
Mobile Computing: Nomadic or Ubiquitous Computing,Applications of Mobile Computing	2
Challenges of Mobile Computing, Mobile Computing Software Platforms, Business Applications of Mobile Computing.	3
Module 5	**12 hrs**
Security and Privacy Issues, Legal Aspects: Mobile Security Concepts, Mobile Security Mechanism	3
Mobile Network Security, Mobile Information Security, Mobile Device Security	2
Legal Aspects: Mobile Device Related Laws	1
Future of Mobile Commerce	1
Mobile Commerce and Consumer Acceptance, Growth of Mobile Value Added Services	3
Mobile Fraud Detection, Future Issues	2

24SJINMCA403	JAVA PROGRAMMING	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		CORE	3	1	0	4

Preamble:	The syllabus is prepared with the view of preparing the MCA Graduates capable of writing readable JAVA programs to solve computational problems that they may have to solve in their professional life. The course content is decided to cover the essential object-oriented programming fundamentals which can be taught within the given slots in the curriculum
Prerequisite:	24SJINMCA105-Introduction to Programming, 24SJINMCA108-Problem Solving and Structured Programming,24SJINMCA205-Introduction to Object Oriented Programming

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the concepts of object-oriented programming paradigms and develop skills in these paradigms using Java	K4
CO2	Apply the concepts of strings, arrays and inheritance	K4
CO3	Analyse and evaluate the use of interfaces, Create Java application programs using sound OOP practices and proper program structuring	K4
CO4	Apply and understand JAVA I/O operations and Web applications using Applets.	K4
CO5	Apply the concepts of Event Handling and provide an exposure to to advanced applications such as TCP and UDP network programming	K4

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2				1	1
CO2	3	3	3					2
CO3	3	2	3	1			1	1
CO4	2	2	3	2	1		1	2
CO5	2	2	2	2	1		1	2

Mark distribution:

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance: 8 marks

Continuous Assessment Test (2 numbers): 20 marks

Assignment/Quiz/Course project: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions Course Outcome 1 (CO1):

Course Outcome 1 (CO1):

1. Categorize constructors in Java
2. Illustrate Java Buzzwords
3. Design a program to illustrate inner class

Course Outcome 2 (CO2):

4. Describe one dimensional array with an example
5. Convince the advantage of 'this' Keyword with proper code sequence
6. Create a program to illustrate the methodology to resolve the overridden method at run time

Course Outcome 3(CO3):

7. Outline the Thread Model in Java
8. Recommend a method to demonstrate the concept of multiple inheritance in java
9. Generate a program to demonstrate user defined exception in Java

Course Outcome 4 (CO4):

10. Define the advantages of streams in java
11. Illustrate Applet skeleton with neat diagram
12. Compose a java program to read characters

Course Outcome 5 (CO5):

13. Explain Delegation event model
14. Compare TCP/IP and Datagram in networking
15. Develop a program to draw a sphere

Syllabus

Module 1

Java Basics: The Java Buzzwords, Data types, Variables, Type conversion and casting, **Introducing Classes-** Class Fundamentals**,** Declaring Objects, Constructors, Overloading methods, Recursion, Introducing Access Control (public, private, protected), Introducing nested and inner classes.

Module 2

Arrays and Strings: One dimensional array, Multidimensional arrays, Exploring String class and methods, String Buffer.

Inheritance Basics: Member access and Inheritance, Using super and this keyword. Method overriding, Dynamic method dispatch, using abstract classes, Using and final Inheritance

Module 3

Packages - Defining a package, finding packages and class path, Importing packages. **Interface**: Defining an interface, implementing Interface.

Exception Handling: Exception handling Fundamentals, using try and catch, Multiple catch clauses, Nested try statements, Throw, throws and Finally, Java's Built-in exceptions, Creating own exception subclasses.

Multithreaded Programming: Java thread Model, creating a Threads - Extending thread and implementing runnable, creating multiple Threads, synchronizing threads (using synchronized statement and synchronized method)

Module 4

I/O Basics, Streams, Byte Streams, Character Streams, reading console inputs, reading characters, Reading Strings

Applets – Applet Basics, Applet Skeleton, Simple Applet display methods, The HTML Applet Tag, passing parameters to applet

Module 5

Event Handling - The Delegation Event Model, Event Classes, Event Listener Interface. Introducing AWT: AWT Classes, Window Fundamentals-Container, Window, Frame, Working with Graphics, Working with Color.

Networking: Networking Basics, client-server, Socket Overview, InetAddress, TCP/IP Client sockets, TCP/IP Server Sockets, Datagrams.

Text Books

1. Herbert Schildt, "The Complete Reference Java 2" , TATA McGRAW-Hill Edition,Fifth Edition,2002.

Reference Books

1. Mahesh P. Matha, "CORE JAVA A Comprehensive Study", PHI Learning Private Limited 2011.
2. Balagurusamy E., "Programming JAVA a Primer", 5/e, McGraw Hill, 2014.

Web Resources

1. https://www.studytonight.com/java/
2. https://www.tutorialspoint.com/java/

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1: Java Basics and Introducing Classes**	**8 hrs.**
1.1	The Java Buzzwords	1
1.2	Data types	1
1.3	Variables	1
1.4	Type conversion and casting	1
1.5	Class Fundamentals, Declaring Objects, Constructors	1
1.6	Overloading methods, Recursion,	1
1.7	Introducing Access Control (public, private, protected)	1
1.8	Introducing nested and inner classes.	1
2	Module 2: Arrays, Strings and Inheritance Basics	**10 hrs.**
2.1	One dimensional array	1

2.2	Multidimensional arrays	1
2.3	Exploring String class and methods	1
2.4	String Buffer	1
2.5	Member access and Inheritance, Using super and this keyword.	1
2.6	Method overriding, Dynamic method dispatch	2
2.7	Using abstract classes	1
2.8	Using and final Inheritance	2
3	**Module 3: Packages, Interface and Exception Handling**	**12 hrs.**
3.1	Defining a package, Finding packages and class path, Importing packages.	2
3.2	Defining an interface	2
3.3	Implementing Interface.	2
3.4	Exception handling Fundamentals, using try and catch, Multiple catch clauses,	2
3.5	Nested try statements, Throw, throws and Finally	2
3.6	Java’s Built-in exceptions, Creating own exception subclasses.	2
4	**Module 4: I/O Basics and Applets**	**8 hrs.**
4.1	Streams, Byte Streams	1
4.2	Character Streams	1
4.3	Reading console inputs, Reading characters	1
4.4	Reading Strings	1
4.5	Applet Basics, Applet Skeleton	1

4.6	Simple Applet display methods	1
4.7	The HTML Applet Tag,	1
4.8	Passing parameters to applet	1
5	Module 5: Event Handling and Networking	10 hrs.
5.1	The Delegation Event Model, Event Classes, Event Listener Interface	1
5.2	Introducing AWT: AWT Classes	1
5.3	Window fundamentals-Container, Window, Frame	1
5.4	Working with Graphics, Working with Color.	1
5.5	Networking Basics, client-server, Socket Overview	1
5.6	InetAddress	1
5.7	TCP/IP Client sockets, TCP/IP ServerSockets	2
5.8	Datagrams.	2

24SJINMCA405	ADVANCED WEB PROGRAMMING	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	With a dynamic learn-by-doing focus, this course encourages the students to explore the designing of web applications by implementing the relevant and recent techniques.
Prerequisite:	24SJINMCA210 Internet Concepts and Web Technology, 24SJINMCA303 User Interface Design

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the fundamental concepts of web architecture, markup languages (HTML), and Cascading Style Sheets (CSS) to design well-structured and visually appealing web pages.	K3
CO2	Apply client-side scripting using JavaScript for creating interactive and dynamic web pages through DOM manipulation, event handling, and form validation.	K3
CO3	Demonstrate knowledge of relational database concepts and perform basic SQL operations (DDL and DML) for data management in web applications.	K3
CO4	Develop dynamic server-side applications using PHP, including database connectivity, data handling, and integration of front-end and back-end components.	K4
CO5	Explain the concepts of distributed computing and web services, including SOAP, WSDL, UDDI, and REST architecture, to understand how interoperable systems communicate over the web.	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	2			1	
CO2	2	3	3	3		1	2	1
CO3	3	3	3	2			1	1
CO4	3	3	3	3		1	2	2
CO5	2	2	3	2	2		3	1

Mark distribution:

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance: 8 marks

Continuous Assessment Test (2 numbers): 20 marks

Assignment/Quiz/Course project: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions Course Outcome 1 (CO1):

Course Outcome 1 (CO1):

1. Define HTML and list its main features.
2. Compare inline, internal, and external style sheets.
3. Develop an HTML form with proper formatting and CSS styling.

Course Outcome 2 (CO2):

4. List the different data types available in JavaScript.
5. Explain the role of the Document Object Model (DOM) in JavaScript.
6. Illustrate how form validation is implemented using JavaScript.

Course Outcome 3(CO3):

7. Write SQL commands to create a table, insert records, and delete rows.
8. Illustrate how the ALTER and TRUNCATE commands are used with examples.
9. Demonstrate a query to retrieve specific data using the SELECT statement with conditions.

Course Outcome 4 (CO4):

10. Explain the steps for setting up a PHP development environment (e.g., XAMPP).
11. Illustrate database connectivity in PHP with an example program**.**
12. Create a PHP program to insert and retrieve records from a MySQL table.

Course Outcome 5 (CO5):

13. Define distributed computing and state its importance.
14. Compare SOAP-based and RESTful web services.
15. Illustrate the basic operational model of web services with a neat diagram.

Syllabus

Module 1

Markup Language (HTML): Formatting and Fonts, Commenting Code, Anchors, Backgrounds, Images, Hyperlinks, Lists, Tables, Frames, HTML Forms. Cascading Style Sheet (CSS): The need for CSS, Basic syntax and structure, Inline Styles, Embedding Style Sheets, Linking External Style Sheets, Backgrounds, Manipulating text, Margins and Padding, Positioning using CSS.

Module 2

Client-Side Scripting using JavaScript: Core features, Data types and Variables, Operators - Expressions and Statements, Functions, Objects, Array, String - Date and Math related Objects, Document Object Model, Event Handling, Form handling and validations.

Module 3

An overview of Relational Database Design: Tables, Attributes, Tuples, Primary keys, Foreign keys, Indexes, DDL Commands – CREATE, ALTER, DROP and TRUNCATE; DML Commands – SELECT, INSERT, UPDATE and DELETE.

Module 4

Server Side Scripting using PHP: Setting up the environment (Example - XAMP server), PHP Programming basics - Print/echo, Variables and constants, Strings and Arrays, Operators, Control structures and looping structures, Functions, Reading Data in Web Pages, Embedding PHP within HTML, establishing connectivity with database, Debugging with phpdbg.

Module 5

Distributed Computing: Importance of Distributed Computing, Client-Server Applications. Web Services: Introduction to Web Services, Motivation and Characteristics, Why Use Web Services, Basic Operational Model of Web Services, Core Web Services Standards: - Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), Web Services Definition Language (WSDL). Universal Description, Discovery, and Integration (UDDI). Introduction to REST: REST Principles, Architecture.

Text Books

1. Steve Prettyman "Learn PHP 7", Object-Oriented Modular Programming using HTML5, CSS3, JavaScript, XML, JSON, and MySQL, Apress, Ist edition(2015).
2. David Flanagan, "JavaScript: The Definitive Guide", 6th Edition", O'Reilly Media.
3. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw Hill Education, 6th Edition (2011).
4. Ludovic Dewailly "Building a RESTful Web Service with Spring", Packt Publishing.(2015).
5. Chris Pitt "Pro PHP MVC", Apress, Ist edition(2012).
6. Steven Holzner, "PHP-The Complete Reference", Tata McGraw Hill, 1st Edition (2007).

Reference Books

1. Thomas A Powell, Fritz Schneider, "JavaScript: The Complete Reference", 3rd Edition, Tata McGraw Hill.
2. Matt Stauffer "Laravel: Up & Running", A Framework for Building Modern PHP Apps, O'Reilly Media, (2016).
3. Athman Bouguettaya, Quan Z. Sheng, Florian Daniel, "Web Services Foundations", Springer (2013).

Web Resources

1. http://php.net/manual/
2. https://www.udemy.com/course/html5-fundamentals-for-beginners/
3. https://www.udemy.com/course/programming-in-javascript/
4. https://www.udemy.com/course/php-mysql-tutorial/
5. https://www.udemy.com/course/jquery-guide/
6. https://www.udemy.com/course/understanding-mvc-model-view-controller-with-php/

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**9 hrs.**
1.1	Markup Language (HTML): Formatting and Fonts, Commenting Code	1

1.2	Anchors, Backgrounds, Images, Hyperlinks	1
1.3	Lists, Tables, Frames, HTML Forms	2
1.4	Cascading Style Sheet (CSS): The need for CSS, Basic syntax and structure	2
1.5	Inline Styles, Embedding Style Sheets, Linking External Style Sheets	2
1.6	Backgrounds, manipulating text, Margins and Padding, Positioning using CSS.	1
2	**Module 2**	**11 hrs.**
2.1	Client-Side Scripting using JavaScript: Core features, Data types and variables	2
2.2	Operators - Expressions and Statements, Functions	2
2.3	Objects, Array, String - Date and Math related Objects	2
2.4	Document Object Model, Event Handling	2
2.5	Form handling and validations	3
3	**Module 3**	**9 hrs.**
3.1	An overview of Relational Database Design: Tables, Attributes	2
3.2	Tuples, Primary keys, Foreign keys	2
3.3	Indexes	1
3.4	DDL Commands – CREATE, ALTER, DROP and TRUNCATE	2
3.5	DML Commands – SELECT, INSERT, UPDATE and DELETE.	2
4	**Module 4**	**11 hrs.**
4.1	Server Side Scripting using PHP: Setting up the environment (Example - XAMP server)	2
4.2	PHP Programming basics - Print/echo, Variables and constants	2
4.3	Strings and Arrays, Operators, Control structures and looping structures	1
4.4	Functions, Reading Data in Web Pages, Embedding PHP within HTML	2
4.5	Establishing connectivity with database, Debugging with phpdbg.	4
5	**Module 5**	**8 hrs.**
5.1	Distributed Computing: Importance of Distributed Computing, Client-Server	

	Applications.	2
5.2	Web Services: Introduction to Web Services, Motivation and Characteristics, Why Use Web Services.	1
5.3	Basic Operational Model of Web Services, Core Web Services Standards: - Extensible Markup Language (XML).	1
5.4	Simple Object Access Protocol (SOAP), Web Services Definition Language (WSDL). Universal Description, Discovery, and Integration (UDDI).	2
5.5	Introduction to REST: REST Principles, Architecture.	2

24SJINMCA407	**ADVANCED SOFTWARE ENGINEERING**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		CORE	3	1	0	4

Preamble:	Most programs on Computer Applications do not give due importance to teaching Software Engineering from an Industry perspective. But this course, built upon the tools and techniques prevalent in Industry today, is supposed to make students Industry-ready.
Prerequisite:	Programming proficiency in at least one C, C++, Java, Python or PHP programming language.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand Software Engineering principles, the SDLC, software models, project planning, and requirements engineering.	K2
CO2	Learn coding standards, documentation, and version control with Git. Ensure software quality across SDLC stages.	K2
CO3	Understand OOP concepts, apply Design Patterns, and write unit tests using frameworks like JUnit or unittest.	K3
CO4	Use Agile and Scrum frameworks. Gain knowledge of software testing, automation, and defect management.	K2
CO5	Manage software configurations, CI/CD pipelines, and deployment automation using tools like Ansible and Robot Framework.	K2

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2	2	1			1
CO2	3	2	2	3	2			2
CO3	3	2	3	3	2			2
CO4	2	3	3	2	3	1	2	2
CO5	2	1	3	3	2	2		2

Mark distribution

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which students should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Course Outcome 1 (CO1)

1. Understand software development as an engineering process and its stages.
2. Gain knowledge of the Software Development Lifecycle (SDLC) and software engineering models.
3. Learn to elicit, model, and document software requirements.
4. Prepare Software Requirements Specifications (SRS) using Use Cases and User Stories.

Course Outcome 2 (CO2)

5. Learn industry-grade coding practices following style guides and coding standards.
6. Work with version control systems, including Git operations like branching, cloning, committing, and conflict resolution.
7. Understand software quality concepts at different SDLC stages with a focus on documentation and conformance.

Course Outcome 3 (CO3)

8. Comprehend Object-Oriented Programming (OOP) principles and apply Design Patterns effectively.
9. Understand and implement Unit Testing using frameworks like JUnit (Java), unit test (Python), or phpdbg (PHP).
10. Gain insights into the xUnit architecture for writing effective tests.
11. Learn the fundamentals of Continuous Integration (CI) and Continuous Delivery (CD) practices.

Course Outcome 4 (CO4)

12. Understand the Agile methodology and implement the Scrum framework for software development.

13. Learn essential software testing methods: black-box testing, white-box testing, and regression testing.
14. Automate testing processes using tools like the Robot Framework to ensure quality software delivery.

Course Outcome 5 (CO5)

15. Gain knowledge of Software Configuration Management (SCM), including dependency and build management.
16. Understand CI/CD pipeline practices for continuous integration and delivery.
17. Use Ansible for deployment automation and Robot Framework for test automation to streamline software releases.

SYLLABUS

Module I (8 Hours)

Introduction to Software Engineering: What is Software Engineering, Characteristics of Software. Life cycle of a software system: software design, development, testing, deployment, Maintenance. Project planning phase: project objectives, scope of the software system, empirical estimation models, COCOMO, staffing and personnel planning.
Software Engineering models: Predictive software engineering models, model approaches, prerequisites, predictive and adaptive waterfall, waterfall with feedback (Sashimi), incremental waterfall, V model; Prototyping and prototyping models.
Software requirements specification, Eliciting Software requirements, Requirement specifications, Software requirements engineering concepts, Requirements modelling, Requirements documentation. Use cases and User stories.

Module II (10 Hours)

Programming Style Guides and Coding Standards; Literate programming and Software documentation; Documentation generators, Javadoc, phpDocumentor.

Version control systems basic concepts: Concept of Distributed version control system and Git; Setting up Git; Core operations in Git version control system using command line interface (CLI): Clone a repository; View history; Modifying files; Branching; Push changes, Clone operation, add, commit, log, diff commands, conflict resolution. Pushing changes to the master; Using Git in IDEs and UI based tools.

Software Quality: Understanding and ensuring requirements specification quality, design quality, quality in software development, conformance quality.

Module III (10 Hours)

OOP Concepts; Design Patterns: Basic concepts of Design patterns, How to select a design pattern, Creational patterns, Structural patterns, Behavioural patterns. Concept of Anti-patterns.

Unit testing and Unit Testing frameworks, The xUnit Architecture, Writing Unit Tests using at least one of Junit (for Java), unittest (for Python) or phpdbg (PHP). Writing tests with Assertions, defining and using Custom Assertions, single condition tests, testing for expected errors, Abstract test.

Module IV (10 Hours)

Concepts of Agile Development methodology; Scrum Framework.

Software testing principles, Program inspections, Program walkthroughs, Program reviews; Blackbox testing: Equivalence class testing, Boundary value testing, Decision table testing, Pairwise testing, State transition testing, Use-case testing; White box testing: control flow testing, Data flow testing.

Testing automation: Defect life cycle; Regression testing, Testing automation; Testing nonfunctional requirements.

Module V (10 Hours)

Software Configuration Management: Using version control, managing dependencies, managing software configuration, Managing build and deployment environments.

Continuous Integration: Prerequisites for continuous integration, Essential practices.

Continuous Delivery: Principles of Software delivery, Introduction and concepts.

Build and deployment automation, Learn to use Ansible for configuration management.

Test automation (as part of continuous integration), Learn to set up test automation cases using Robot Framework.

Notes

1. At the end of Module 1, conduct the following class work with appropriate evaluation points:

   Prepare Software Specification Document for a moderately complex process flow system (e.g.

   Broadband fault booking and resolution system covering technical, operational and commercial aspects, covering organizational and subscriber use cases).

2. At the end of Module 2, clone an open source project using Git and perform all based operations.

Web-based Resources

1. Git Handbook https://guides.github.com/introduction/git-handbook/ Retrieved 8 July 2020

   [Module 2]

2. Git User Manual https://mirrors.edge.kernel.org/pub/software/scm/git/docs/user-manual.html

   Retrieved 8 July 2020 [Module 2]

3. Introduction to Software Engineering/Quality
   https://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Quality Retrieved 8 July2020 [Module 2]
4. unittest — Unit testing framework https://docs.python.org/3/library/unittest.html Retrieved 8 July 2020 [Module 4]
5. What is CI/CD? https://www.redhat.com/en/topics/devops/what-is-ci-cd Retrieved 8 July 2020 [Module 5]

References

1. Philip A. Laplante, What Every Engineer Should Know about Software Engineering, CRC Press [Module 1]
2. Murali Chemuturi, Mastering Software Quality Assurance: Best Practices, Tools and Technique for Software Developers, J Ross Publishing [Module 2]
3. Ben Straub, Scott Chacon, Pro Git, 2nd Edition, Apress [Module 2]
4. Erich Gamma et. al., Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley [Moule 3]
5. Vaskaran Sarcar, Java Design Patterns: A Hands-On Experience with Real-World Examples, Apress [Module 3]
6. Alistair Cockburn and Robert Cecil Martin, Agile Software Development: The Cooperative Game (2nd edition), Addition Wesley [Module 4]
7. Ken Schwaber , Agile Software Development with Scrum, Pearson [Module 4]
8. Lisa Crispin, Agile Testing: A Practical Guide for Testers and Agile Teams, Adison Wesley
9. Paul Hamill, Unit Test Frameworks, O'Reilly Media [Module 4]
10. Glenford J. Myers, et. al., The Art of Software Testing, Wiley [Module 4, 5]
11. Lee Copeland, A Practitioner's Guide to Software Test Design, Artech House Publishers [Module 4, 5]
12. Jez Humble and David Farley, Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Pearson Education [Module 5]

Course Contents and Lecture Schedule

No	**Topic**	**No. of Lectures**
1	**Module 1- Software Engineering**	**8**
1.1	What is Software Engineering, Characteristics of Software Engineering.	1
1.2	Life cycle of a software system	1
1.3	Project planning	1
1.4	Software Engineering Models	2

1.5	Software Requirements Specification	3
	Module 2- Industry Best Practices	**10**
2.1	Programming style guides and coding standards	1
2.2	Software version control systems, basic concepts	1
2.3	Git distributed version control system, introduction	2
2.4	Common operations in Git	4
2.5	Software quality, achieving	2
	Module 3- System Design Methodologies	**10**
3.1	Object Oriented Programming	1
3.2	Software Design Patterns	4
3.3	Unit Testing concepts and xUnit architecture	1
3.4	Unit testing frameworks: Junit, unittest, phpdbg	2
3.5	Writing unit test code	2
	Module 4- Agile Development Methodology	**10**
4.1	Agile Development methodology, introduction	2
4.2	Scrum framework	5
4.3	Automated testing	3
	Module 5- Continuous Integration and Continuous Developmen (CI/CD)	**10**
5.1	Configuration Management	2
5.2	Continuous Integration, concepts and practices	2
5.3	Continuous Delivery, concepts and practices	2
5.4	Build and deployment automation	2
5.5	Test automation for CI/CD	2

24SJINMCA409	DISTRIBUTED COMPUTING	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course provides a solid theoretical foundation that furnishes the student with insight into the more sophisticated distributed computing systems and understand large data processing facilities that provides significant computing capabilities.
Prerequisite:	24SJINMCA206 Operating Systems

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Recognize distributed systems architecture.	K4
CO2	Distinguish between various communication models in distributed systems	K2
CO3	Examine on various consistency models	K2
CO4	Examine different cloud models and services	K4
CO5	Understand and analyse different virtualisation mechanisms	K4

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1		2		1			2	
CO2				1				
CO3	1	1	1	2		1		
CO4					2			2
CO5		1			2			

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 hours

Continuous Internal Evaluation Pattern

Attendance : 8 marks

Continuous Assessment Test : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment

Course Outcome 1 (C01):

1. List out the goals of a distributed computing system
2. Describe code migration in detail.
3. Summarize on various types of distributed computing system.

Course Outcome 2 (CO2):

4. Explain one election algorithm in detail
5. With necessary diagrams, explain RPC
6. Explain one mutual exclusion algorithm in detail

Course Outcome 3 (CO3):

7. Describe various consistency models
8. Explain process resilience.
9. Explain reliable client-server communication

Course Outcome 4 (CO4):

10. With a proper diagram, explain NIST cloud computing reference architecture.
11. Differentiate b/w public and private cloud
12. Summarize on various cloud models.

Course Outcome 5 (CO5):

13. Assess the various implementation levels of virtualization
14. Explain layered cloud architecture development
15. Summarize on virtualization of CPU, Memory and I/O Devices.

Syllabus

Module 1

Introduction to distributed systems – definition, goals, types (Distributed Computing Systems, Distributed Information Systems, Distributed Pervasive Systems (Home Systems)). **Processes** – Virtualisation, Clients, Servers (General Design Issues, General Organization, Distributed Servers), Code Migration (Approaches to Code Migration, Migration and Local Resources)

Module 2

Communication – RPC (Basic RPC Operation-Conventional Procedure Call, Client and Server Stubs, Parameter Passing-Passing Value Parameters, Asynchronous RPC).

Synchronization – Clock synchronization (Physical Clocks, Global Positioning System), Election Algorithms (Bully Algorithm, Ring Algorithm), Mutual Exclusion (Centralized Algorithm, Distributed Algorithm, Token Ring Algorithm)

Module 3

Consistency and replication – Data centric consistency, client centric consistency. **Fault Tolerance** – introduction, process resilience, reliable client-server communication, reliable group communication (Basic Reliable-Multicasting Schemes, Atomic Multicast-Virtual Synchrony).

Module 4

Technologies for Network-Based System – System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture. **Cloud Models and Services** – Cloud models (IaaS, PaaS, and SaaS) – Public vs Private Cloud

Module 5

Virtual machines and Virtualisation of clusters and data centres: Implementation Levels of Virtualization, Virtualization Structures /Tools and Mechanisms, Virtualization of CPU, Memory and I/O Devices.

Architectural Design of Compute and Storage Clouds: Layered Cloud Architecture Development – Architectural Design Challenges.

Reference Books

1. Distributed Systems – Principles and Paradigm, Tanenbaum& Van Steen, 2nd Edition, PHI Publications
2. "Cloud Computing: Implementation, Management, and Security" John W.Rittinghouse and James F.Ransome, CRC Press, 2010.
3. "Cloud Computing, A Practical Approach" Toby Velte, Anthony Velte, Robert Elsenpeter, TMH, 2009.
4. "Cloud Computing – insights into New-Era Infrastructure", Kumar Saurabh, Wiley India, 2011.
5. "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Morgan Kaufmann Publishers, 2012.

MOOC Courses

1. https://www.coursera.org/learn/cloud-computing?specialization=cloud-computing
2. https://www.coursera.org/learn/cloud-computing-2?specialization=cloud-computing

Web Resources

1. https://www.tutorialspoint.com/Distributed-Systems

Course Contents and Lecture Schedule

No	Topic	No. of lectures
1	**Module 1**	**10 hrs.**
1.1	Definition	1
1.2	Goals	1
1.3	Types	2
1.4	Virtualization	1
1.5	Clients	1
1.6	Servers	2
1.7	Code Migration	2
2	**Module 2**	**8 hrs.**
2.1	RPC	2
2.2	Clock synchronization	2
2.3	Election Algorithms	2
2.4	Mutual Exclusion	2
3	**Module 3**	**10 hrs.**
3.1	Data centric consistency	1
3.2	Client centric consistency	1
3.3	Fault Tolerance – introduction	2

3.4	Process resilience	2
3.5	Reliable client-server communication	2
3.5	Reliable group communication	2
4	**Module 4**	**11 hrs.**
4.1	Technologies for Network-Based System	2
4.2	System Models for Distributed and Cloud Computing	3
4.3	NIST Cloud Computing Reference Architecture	3
4.4	Cloud Models and Services	3
5	**Module 5**	**9 hrs.**
5.1	Implementation Levels of Virtualization	1
5.2	Virtualization Structures /Tools and Mechanisms	2
5.3	Virtualization of CPU, Memory and I/O Devices	2
5.4	Layered Cloud Architecture Development	2

24SJINMCA431	Java Programming - Lab	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	1

Preamble:	The syllabus is prepared with the view of preparing the MCA Graduates capable of writing Java programs to solve computational problems that they may have to solve in their professional life. The students can explore various programming constructs and data structures for the basic problem solving in Java language. The course content is decided to cover the essential programming fundamentals.
Prerequisite:	24SJINMCA132: Problem Solving and Structured Programming Lab, 24SJINMCA233: Basic Object Oriented Programming Lab

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the concepts of object-oriented programming paradigms and develop skills in Java.	K4
CO2	Apply the concepts of constructors and destructors.	K4
CO3	Implement various Inheritance concepts.	K4
CO4	Analyse and evaluate the use of interfaces, Exception Handling, and Multithreaded Programming.	K4
CO5	Apply and understand JAVA I/O operations, Applets, and Event Handling concept.	K4

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	1	1	1	2
CO2	3	3	2	2	1	1	1	2
CO3	3	2	3	3	2	1	1	2
CO4	2	2	3	3	2	1	1	2
CO5	2	2	3	2	2	1	1	2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day-to-day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 marks</td></tr>
<tr><td rowspan="4">Problem solving

(Based on difficulty level, one or more questions may be given)</td><td>Flowchart/ Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output and Pushing to remote Git repository</td><td>20%</td></tr>
<tr><td colspan="3">Total Marks</td><td>50 marks</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Create a program to check whether the given number is prime or not using class.
2. Create a program to prepare a Student Record using class and object to, computing total marks and displaying the results.

Course Outcome 2 (CO2)

3. Create a program to demonstrate a copy constructor

Course Outcome 3(CO3):

4. Create a program to demonstrate single and multilevel Inheritance.
5. Recommend a method to demonstrate the concept of multiple inheritance in java

Course Outcome 4 (CO4):

6. Create a program to demonstrate Interface
7. Generate a program to demonstrate user defined exception in Java

Course Outcome 5 (CO5):

8. Develop a program to draw a rectangle

Syllabus

Classes: Class Fundamentals, Declaring Objects, Constructors, Introducing Access Control (public, private, protected)

Inheritance: Member access and Inheritance, Using super and this keyword. Method overriding, Dynamic method dispatch, Using abstract classes, Using and final Inheritance

Packages: Defining a package, Importing packages.

Interface: Defining an interface, implementing Interface.

Multithreaded Programming: Creating a Threads - Extending thread and implementing runnable, Creating multiple Threads

I/O: Byte Streams, Character Streams, Reading console inputs, Reading characters, Reading Strings

Applets: Applet, passing parameters to applet

Event Handling: Event Listener Interface. AWT Classes, Working with Graphics, Working with Color.

Networking: TCP/IP Client sockets, TCP/IP Server Sockets, Datagrams.

Reference Books

1. Herbert Schildt, “The Complete Reference Java 2” , TATA McGRAW-Hill Edition,Fifth Edition,2002.

2. Mahesh P. Matha, “CORE JAVA A Comprehensive Study”, PHI Learning Private Limited 2011.
3. Balagurusamy E., “Programming JAVA a Primer”, 5/e, McGraw Hill, 2014.

Web Resources

- https://www.tutorialspoint.com/java/
- https://dev.java/learn/getting-started/
- https://docs.oracle.com/javase/tutorial/

List of Lab Experiments/Exercises

1. Write a Java program that takes three numbers as input to calculate and print the average of the numbers.

2. Create a class Average, the methods are void getInput(Use Scanner class to accept Input),void add(), double average(). Create a class AverageDemo and create objects of Average class and call the above methods.

3. Write a Java program to print the area and perimeter of a Rectangle. Also implement the concept of **this** keyword.

4. Create a class Rectangle, the methods are
   - double areaRect(double x,double y) (area=w*l)
   - double periRect(double x,double y) (perimeter=2*(l+w))

   create a class RectangleDemo and create objects of class Rectangle and call the above methods. (Use Scanner class to accept Input)

5. Create a java program to implement constructor Overloading

6. Create a java program that displays area of different figers, (Rectangle,Square,Triangle) using method overloading. Methods are
   - Void area(int,int)
   - Void area(int)
   - Void area(double,double)

7. Create a java program to perform Matrix addition
8. WAP to create a package for Book details giving BookName, AuthorName, Price and year_of_publishing. Implement this package for the creation and use of another class Library

9. Create a java program to implement multiple inheritance using Interface
10. Develop a program to demonstrate Arithmetic Exception
11. Develop a program to demonstrate try with multiple catch statements
12. Create a Program to throw a user defined exception if the product of two integers is greater than 50
13. Create an applet Program to demonstrate the life cycle of applet
14. Create a program to embed image in applet
15. Create a program to embed sound in an applet
16. Create an applet program to draw an Indian flag
17. Create a java Program to implement client and server communication using TCP
18. Create a java Program to implement client and server communication using UDP

Course Contents and Lecture Schedule

No	Topic	No of Hours
1	**Classes**: Class Fundamentals , Declaring Objects, Constructors, Introducing Access Control (public, private, protected)	9
2	**Inheritance:** Member access and Inheritance, Using super and this keyword. Method overriding, Dynamic method dispatch, Using abstract classes, Using and final Inheritance	7
3	**Packages** : Defining a package, Importing packages.	7
4	**Interface**: Defining an interface, implementing Interface.	5
5	**Multithreaded Programming**: Creating a Threads - Extending thread and Implementing runnable ,Creating multiple Threads	5
6	**I/O:** Byte Streams, Character Streams, Reading console inputs, Reading characters, Reading Strings	5

7	**Applets:** Applet, passing parameters to applet	5
8	**Event Handling:** Event Listener Interface. AWT Classes, Working with Graphics, Working with Color. **Networking** TCP/IP Client sockets, TCP/IP Server Sockets, Datagrams.	5

24SJINMCA433	ADVANCED WEB PROGRAMMING LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble:	With a dynamic learn-by-doing focus, this laboratory course encourages the students to explore the designing of web application by implementing the relevant and recent techniques. This course challenges the students to exercise their creativity in both programming and designing.
Prerequisite:	Basic understanding of computer programming, Internet and Database etc. is very helpful.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Explore markup languages features and create interactive web pages using them.	K2
CO2	Learn and design client-side validation using scripting languages.	K2
CO3	Design front end web page and connect to the back-end databases.	K3
CO4	Do Client-side & Server-side scripting	K3
CO5	Develop Web Applications	K3

Mapping of course outcomes with program outcomes								
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	1	2					
CO2	3	2	3					
CO3	3		2					
CO4	3	1	2					
CO5	3	2	2	1				

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%

Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 marks</td></tr>
<tr><td rowspan="4">Problem solving
(Based on difficulty level, one or more questions may be given)</td><td>Flowchart/ Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output and Pushing to remote Git repository</td><td>20%</td></tr>
<tr><td colspan="3">Total Marks</td><td>50 marks</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Model a simple HTML file to demonstrate the use of different tags.
2. Create a HTML file to link to different HTML page which contains images, tables, and also link within a page.
3. Create a HTML page with different types of frames such as floating frame, navigation frame & mixed frame.
4. Analyze CSS by applying the different styles using inline, external & internal style sheets in a HTML file.
5. Demonstrate a registration form using HTML.

Course Outcome 2 (CO2)

6. Create a HTML page to explain the use of various predefined functions in a string and math object in java script.
7. Generate the calendar using JavaScript code by getting the year from the user.
8. Create a HTML registration form and to validate the form using JavaScript code.
9. Evaluating JavaScript Event Handling for every click of a button to change the background color of a HTML page.

10. Create a HTML page to display a new image and text when the mouse comes over the existing content in the page using JavaScript Event Handling.
11. Create a HTML page to show online exam using JavaScript.

Course Outcome 3(CO3):

12. Develop a PHP program to connect to a database and retrieve data from a table and show the details in a neat format.

Course Outcome 4 (CO4):

13. Outline a registration form using PHP and do necessary validations.
14. Compose Electricity bill from user input based on a given tariff using PHP.
15. Build a PHP code to store name of students in an array and display it using print_r function. Sort and Display the same using asort & arsort functions.
16. Build a PHP code to store name of Indian Cricket players in an array and display the same in HTML table.

Course Outcome 5 (CO5):

17. Develop Web applications using HTML and PHP and deploy.
18. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.
19. Develop a web application for Airline Reservation System using any PHP framework (Laravel, CodeIgniter, Symfony, CakePHP etc.).
20. Test the application on an Application Server.

Syllabus

Introduction To Web: Client/Server concepts, Components of Web Application, Types of Web Content, Overview of HTTP - HTTP request – response, Generation of dynamic web pages, Application Servers, Web Security.

Markup Language (HTML): Formatting and Fonts, Commenting Code, Anchors, Backgrounds, Images, Hyperlinks, Lists, Tables, Frames, HTML Forms.

Cascading Style Sheet (CSS): The need for CSS, Basic syntax and structure, Inline Styles, Embedding Style Sheets, Linking External Style Sheets, Backgrounds, Manipulating text, Margins and Padding, Positioning using CSS.

Client Side Scripting using JavaScript: Core features, Data types and Variables, Operators - Expressions and Statements, Functions, Objects, Array, String - Date and Math related Objects, Document Object Model, Event Handling, Form handling and validations.

An overview of Relational Database Design: Tables, Attributes, Tuples, Primary keys, Foreign keys, Indexes, DDL Commands – CREATE, ALTER, DROP and TRUNCATE; DML Commands – SELECT, INSERT, UPDATE and DELETE.

Server Side Scripting using PHP: Setting up the environment (Example - XAMP server), PHP Programming basics - Print/echo, Variables and constants, Strings and Arrays, Operators, Control structures and looping structures, Functions, Reading Data in Web Pages, Embedding PHP within HTML, Establishing connectivity with database, Debugging with phpdbg.

Web Application development in any PHP framework (Laravel, CodeIgniter, Symfony, CakePHP etc.): Naming convention, MVC model, Connectivity with Database, Database interaction.

Debugging web apps: Browser debugging tools (Any browser web developer tools) – View and change the DOM and CSS, Console, Debug JavaScript, View and debug network activity, Performance tools etc.

Reference Books

1. David Flanagan, "***JavaScript: The Definitive Guide***", 6th Edition", O'Reilly Media
2. Douglas E Comer, "***The Internet Book: Everything You Need to Know About Computer Networking and How the Internet Works***", 4th Edition, Prentice Hall
3. Harvey Deitel and Abbey Deitel, "***Internet and World Wide Web - How To Program***", 5th Edition, Pearson Education
4. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "***Database System Concepts***", McGraw Hill Education, 6th Edition (2011)
5. Steve Suehring, Tim Converse, and Joyce Park, "***PHP6 and MySQL Bible***", Wiley India Pvt Ltd (2009)
6. Steven Holzner, "***PHP-The Complete Reference***", Tata McGraw Hill, 1st Edition (2007)
7. Thomas A Powell, Fritz Schneider, "***JavaScript: The Complete Reference***", 3rd Edition, Tata McGraw Hill

Web Resources

1. http://php.net/manual/
2. https://www.udemy.com/course/html5-fundamentals-for-beginners/
3. https://www.udemy.com/course/programming-in-javascript/
4. https://www.udemy.com/course/php-mysql-tutorial/

List of Lab Experiments/Exercises

1. Create a simple HTML file to demonstrate the use of different tags.
2. Create a HTML file to link to different HTML page which contains images, tables, and also link within a page.

3. Create a HTML page with different types of frames such as floating frame, navigation frame & mixed frame.
4. Create a HTML file by applying the different styles using inline, external & internal style sheets.
5. Create a registration form using HTML.
6. Create a HTML page to explain the use of various predefined functions in a string and math object in java script.
7. Generate the calendar using JavaScript code by getting the year from the user.
8. Create a HTML registration form and to validate the form using JavaScript code.
9. Create a HTML page to change the background color for every click of a button using JavaScript Event Handling.
10. Create a HTML page to display a new image and text when the mouse comes over the existing content in the page using JavaScript Event Handling.
11. Create a HTML page to show online exam using JavaScript.
12. Develop a registration form using PHP and do necessary validations.
13. Compose Electricity bill from user input based on a given tariff using PHP.
14. Build a PHP code to store name of students in an array and display it using print_r function. Sort and Display the same using asort & arsort functions.
15. Build a PHP code to store name of Indian Cricket players in an array and display the same in HTML table.
16. Develop a PHP program to connect to a database and retrieve data from a table and show the details in a neat format.
17. Develop Web applications using HTML and PHP and deploy.
18. Using PHP and MySQL, develop a program to accept book information viz.Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.
19. Develop a web application for Airline Reservation System using any PHP framework (Laravel, CodeIgniter, Symfony, CakePHP etc.).
20. Test the application on an Application Server.

Course Contents and Lecture Schedule

No	Topic	No. of lectures
1	Client/Server concepts, Components of Web Application, Types of Web Content, Overview of HTTP - HTTP request – response, Generation of dynamic web pages, Application Servers, Web Security.	1
2	HTML - Formatting and Fonts, Commenting Code, Anchors, Backgrounds, Images, Hyperlinks.	4
3	HTML - Lists, Tables, Frames, HTML Forms.	4
4	The need for CSS, Basic syntax and structure, Inline Styles, Embedding Style Sheets, Linking External Style Sheets, Backgrounds.	4
5	CSS - Manipulating text, Margins and Padding, Positioning using CSS.	4

6	JavaScript: Core features, Data types and Variables, Operators - Expressions and Statements.	3
7	JavaScript: Functions, Objects, Array, String - Date and Math related Objects, Document Object Model, Event Handling.	4
8	JavaScript: Form handling and validations.	4
9	An overview of Relational Database Design: Tables, Attributes, Tuples, Primary keys, Foreign keys, Indexes, DDL Commands – CREATE, ALTER, DROP and TRUNCATE.	4
10	DML Commands – SELECT, INSERT, UPDATE and DELETE.	4
11	PHP: Setting up the environment (Example - XAMP server), PHP Programming basics - Print/echo, Variables and constants.	4
12	Strings and Arrays, Operators, Control structures and looping structures.	4
13	Functions, Reading Data in Web Pages, Embedding PHP within HTML, Establishing connectivity with database.	4
14	PHP framework: naming convention, MVC model, Connectivity with Database, Database Interaction.	6

ST.JOSEPH'S COLLEGE OF ENGINEERING AND TECHNOLOGY, - PALAI -

AUTONOMOUS

MCA Integrated

SEMESTER 8

24SJINMCA402	ADVANCED DATABASE MANAGEMENT SYSTEMS	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course provides the basic concepts and terminology related to relational and non- relational database management systems. The concept of advanced DBMS techniques and new generation databases like MongoDB, HBase and Cassandra are also introduced. This course serves as a prerequisite for many advanced courses in Data Science and Machine Learning areas.
Prerequisite:	Basic knowledge in Database Management Systems.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the fundamentals of relational database systems including data models, database architectures and ER features.	K3
CO2	Analyze and apply the different normalization techniques.	K4
CO3	Assess the basic issues of transaction processing and concurrency control.	K4
CO4	Understand the roles that databases play in organizations and familiarize with basic database storage, file organization, database accessing techniques.	K4
CO5	Understand the basics of query processing, object-oriented and distributed databases, and analyze non-relational database systems, structures, and XML.	K4

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1				1			1	
CO2	3	3	3	2			2	2
CO3	1	2	2	2		2		
CO4					1		1	
CO5	2			1				

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks

Continuous Assessment Test (2 numbers) : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1)

1. Examine why databases are important.
2. Describe the basic features of the relational data model and discuss their importance to the end user and the designer.
3. Analyze the graphic depiction of relationships among the entities and examine how these depictions help in the database design process.

Course Outcome 2 (CO2)

1) Evaluate and design good table structures to control data redundancies and anomalies.

Course Outcome 3(CO3)

1. Explain the database transaction and its properties.
2. Describe concurrency control and analyze the role it plays in maintaining the database integrity.
3. Assess the common algorithms for concurrency control.
4. Define deadlock and discuss the strategies for managing deadlocks.
5. Examine how database recovery management is used to maintain database integrity.

Course Outcome 4 (CO4)

1. Discuss the various disk-organization techniques.
2. Describe the various data structures that allow fast access to data.
3. Analyze and examine the different indexing techniques.

Course Outcome 5 (CO5)

1. Describe the basics of query processing and evaluate the query processing cost.
2. Analyze the concept of object-oriented databases and distributed databases.
3. Explain the concept of XML.
4. Describe the various NoSQL databases.

SYLLABUS

Module I:

Relational Databases:- IntroductE-R Diagrams: Basic structure, Complex attributes, Roles, Non binary relationship sets, Weak Entity Set,ion - Purpose of Database System – Database System Applications - View of data: Data Abstraction, Instances and Schemas, Data Models – Database Architecture - Database Users and Administrators: Database Users and Interfaces, DBA – Introduction to the Relational Model: Structure of Relational Database, database Schema, Keys, Relational Query language – The Relational Algebra: Fundamental Operations, Formal definition of the relational algebra, additional relational algebra operations – The Entity- Relationship model: Entity Set, Relationship Set, Attributes – Constraints: Mapping cardinalities, Key Constraints, Participation Constraints - Relational Database Design using ER- to Relational Mapping – Extended ER Features: Specialization, Generalization, Attribute inheritance, Constraints on generalization, Aggregation.

Module II:

Database Design:- Database Tables and Normalization – The Need for Normalization – The Normalization Process: Inference Rules for Functional Dependencies (proof not needed) - Minimal set of Functional Dependencies - Conversion to First Normal Form, Conversion to Second Normal Form, Conversion to Third Normal Form - Improving the Design - Surrogate

Key Considerations - Higher Level Normal Forms: Boyce/Codd Normal Form, Fourth Normal Form, Join dependencies and Fifth Normal Form – Normalization and Database Design.

Module III:

Transaction Management and Concurrency Control:- Transaction: Evaluating Transaction Results, Transaction Properties, Transaction Management with SQL, The Transaction Log – Concurrency Control: Lost Updates, Uncommitted Data, Inconsistent Retrievals, The Scheduler – Concurrency Control with Locking Methods: Lock Granularity, Lock Types, Two Phase Locking to Ensure Serializability, Deadlocks – Concurrency Control with Timestamping Methods: Wait/Die and Wait/Wound Schemes – Concurrency Control with Optimistic Methods- Database Recovery Management: Transaction Recovery.

Module IV:

Data Storage and Querying:- RAID – File Organization – Organization of Records in Files – Indexing and Hashing: Basic concept, Ordered Indices, B+ tree Index Files: Structure of a B+-Tree (structure only, algorithms not needed) - B tree index files – Static Hashing – Dynamic Hashing – Query Processing: Overview - Selection Operation.

Module V:

System Architecture, Object Oriented Databases, XML and NoSQL:- Distributed Databases: Homogeneous and Heterogeneous Databases, Distributed Data Storage,

Distributed Transactions - Object Based Databases: Overview, Complex Data types, Structured types and inheritance in SQL, Table Inheritance, Array and Multiset types in SQL, Object identity and reference types in SQL - XML: DTD and XML Schema, XML presentation, XML Applications - Next Generation Databases: Distributed Relational Databases - Nonrelational Distributed Databases - MongoDB Sharding and Replication - Hbase - Cassandra - CAP Theorem.

Text Books

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan," Database System Concepts", McGraw Hill Education, 6th Edition, 2011. (for Module 1 Refer Chapter 1 [1.1 to 1.3, 1.9,1.12], Chapter 2 [2.1-2.3,2.5], Chapter 6 [6.1], Chapter 7 [7.2, 7.3, 7.8(7.81. To 7.8.5)], for Module 4 Refer Chapter 10 [10.3, 10.5, 10.6], Chapter 11 [11.1, 11.2,

11.3(11.3.1), 11.4.5 and module 5 Refer Chapter 19 [19.1,19.2, 19.3 - Distributed Databases], Refer Chapter 22 [22.1 to 22.6 - Object Based Databases]).

2. Ramez Elmasri, Shamkant B.Navathe, " Fundamentals of Database Systems ", Pearson Education, 5th Edition, 2007. (for Module 1 - Refer Chapter 7 [7.1] - 7.1.1 - Relational Database Design using ER- to Relational Mapping]) and for Module 2 - Refer Chapter 10 [10.2.2 and 10.2.4], Refer Chapter 11 [11.4 - Join dependencies and Fifth Normal Form).
3. Guy Harrison, "Next Generation Databases: NoSQL, NewSQL, and Big Data", Generation Databases and CAP Theorem).
4. Rob, Peter and Carlos Coronel, "Database Principles: Fundamentals of Design, Implementation and Management", 9th Edition, 2011. (for Module 2, refer chapter 6) and (for module 3, refer chapter 10) and (for Module 5, refer Chapter 14 -XML).

Reference Books

1. Ashutosh Kumar Dubay, "Database Management Concepts", S.K. Kataria & Sons, 1st Edition (2012).
2. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", McGraw Hill, 3rd Edition (2014).
3. Thomas M Connolly and Carolyn E Begg, "Database systems- A Practical Approach to Design, Implementation and Management", Pearson Education, 4th Edition (2014).

Web Resources

1. Introduction to Databases (nptel) https://nptel.ac.in/courses/106106220
2. Database Design (nptel) https://nptel.ac.in/courses/106105175
3. Introduction to Database Systems and Design https://pll.harvard.edu/course/cs50s-introduction-databases-sql
4. Fundamentals of Database Systems https://www.coursera.org/learn/introduction-to-databases
5. Database Management Essentials (Coursera) https://www.coursera.org/learn/database-management

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Module I: Relational Databases**	**15 hrs**
1.1	Introduction - Purpose of Database System - Database System Applications	1
1.2	View of data: Data Abstraction, Instances and Schemas, Data Models	1
1.3	Database Architecture	1
1.4	Database Users and Administrators: Database Users and Interfaces, DBA	1
1.5	Introduction to the Relational Model: Structure of Relational Database, database Schema, Keys, Relational Query language	1
1.6	The Relational Algebra: Fundamental Operations, Formal definition of the relational algebra, additional relational algebra operations	2
1.7	The Entity-Relationship model: Entity Set, Relationship Set,Attributes	1
1.8	Constraints: Mapping cardinalities, Key Constraints, Participation Constraints	2
1.9	E-R Diagrams: Basic structure, Complex attributes, Roles, Non binary relationship sets, Weak Entity Set	1
1.10	Relational Database Design using ER- to Relational Mapping	2
1.11	Extended ER Features: Specialization, Generalization, Attribute inheritance, Constraints on generalization, Aggregation.	2
2	**Module II: Database Design**	**9 hrs**
2.1	Database Tables and Normalization - The Need for Normalization	1
2.2	The Normalization Process: Inference Rules for Functional Dependencies (proof not needed) - Minimal set of Functional Dependencies - Conversion to First Normal Form, Conversion to Second Normal Form	2
2.3	Conversion to Third Normal Form	1
2.4	Improving the Design - Surrogate Key Considerations	1

2.5	Higher Level Normal Forms: Boyce/Codd Normal Form	1
2.6	Fourth Normal Form	1
2.7	Join dependencies and Fifth Normal Form	1
2.8	Normalization and Database Design	1
3	**Module III: Transaction Management and Concurrency Control**	**9 hrs**
3.1	Transaction: Evaluating Transaction Results, Transaction Properties	1
3.2	Transaction Management with SQL, The Transaction Log	1
3.3	Concurrency Control: Lost Updates, Uncommitted Data, Inconsistent Retrievals, The Scheduler	2
3.4	Concurrency Control with Locking Methods: Lock Granularity	1
3.5	Lock Types, Two Phase Locking to Ensure Serializability	1
3.6	Deadlocks	1
3.7	Concurrency Control with Timestamping Methods: Wait/Die and Wait/Wound Schemes, Concurrency Control with Optimistic Methods, Database Recovery Management: Transaction Recovery	2
4	**Module IV: Data Storage and Querying**	**10 hrs**
4.1	RAID	1
4.2	File Organization	1
4.3	Organization of Records in Files	1
4.4	Indexing and Hashing: Basic concept, Ordered Indices	1
4.5	B+ tree Index Files: Structure of a B+-Tree, B tree Index Files	2
4.6	Static Hashing, Dynamic Hashing	2
4.7	Query Processing: Overview, Selection Operation	2

5	**Module V: System Architecture, Object Oriented Databases, XML and NoSQL**	**13 hrs**
5.1	Distributed Databases: Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions	2
5.2	Object Based Databases: Overview, Complex Data types	1
5.3	Structured types and inheritance in SQL	1
5.4	Table Inheritance	1
5.5	Array and Multiset types in SQL	1
5.6	Object identity and reference types in SQL	1
5.7	XML: DTD and XML Schema	1
5.8	XML presentation, XML Applications	1
5.9	Next Generation Databases: Distributed Relational Databases - CAP Theorem	1
5.10	Non-relational Databases – MongoDB Sharding and Replication	1
5.11	HBase	1
5.12	Cassandra	1

24SJINMCA404	ADVANCED COMPUTER NETWORKS	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course intends to provide insight into Advanced Computer Networks. A software professional should have an understanding of layered network architecture. Various kinds of network architectures, issues in integrating networks to modern application development are to be addressed. It is also intended to expose the student to modern technologies such as IPV6 and software defined networks. More detailed treatment can be done through seminars, assignments and talks by eminent external experts.
Prerequisite:	Basic concepts of computer systems.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Comprehend the terminology and concepts of basic communication model, analyze the protocol layers and design application layer protocols.	K2
CO2	Understand and analyze the various transport layer protocols.	K2
CO3	Compare and contrast various routing algorithms in the network layer.	K2
CO4	Understand and analyze the concepts of link layer and physical layer.	K2
CO5	Understand how modern cellular and wireless networks work.	K2

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	1			1	1
CO2	3	3	2					
CO3	3	3	2					
CO4	3	2	2	1				
CO5	3	1	2	1				

Mark distribution

Continuous Internal Evaluation Pattern:

Attendance: 8 marks

Continuous Assessment Test (2numbers): 20 marks

Assignment/Quiz/Course project: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question.

Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Outcome (CO1):

1. Explain HTTP request-response behavior with a neat diagram.
2. Compare and contrast OSI and TCP/IP network reference models.
3. Explain the importance of layering in data communication.

Course Outcome 2 (CO2):

1. Explain the process of three-way handshaking in TCP.
2. Compare and contrast Multiplexing and De-multiplexing process in transport layer.
3. Explain How TCP is controlling congestion during data transmission

Course Outcome 3 (CO3):

1. Explain how multicast routing is used in routing protocols.
2. Compare and contrast IPV4 and IPV6.
3. Differentiate virtual circuits and datagram networks.

Course Outcome 4 (CO4):

1. Explain how parity is used to achieve error detection in data communication.
2. Illustrate IEEE 802.3 frame structure.
3. Write short notes on routers, switches and bridges.

Course Outcome 5 (CO5):

1. List out and explain the various IEEE 802.11 WLAN Components.
2. Explain the architecture of Bluetooth in personal area networks.
3. Explain any six network attacks and their counter measures.

Syllabus

Module I (10 Hours)

Overview of Computer Networks and the Internet History, Protocols, Review of last mile technologies used for internet access, Packet switching, Basic ideas about delay, queuing, throughput, Concept of Quality of Service, Protocol layering, OSI model and TCP model.

Module II (10 Hours)

Transport Layer Protocols: Introduction to transport layer, Multiplexing and de-multiplexing, Principles of Reliable data transfer - Stop-and-wait and Go-back- N design and evaluation, Connection oriented transport TCP, Connectionless transport UDP, Principles of congestion control -efficiency and fairness.

Module III (10 Hours)

Network Layer Protocols: Virtual circuits and datagrams, Principles of routing, internet protocol Ipv4, CIDR Routing algorithms: Link-state and distance vector routing, Routing on the internet RIP, OSPF and BGP, Multicast routing, Introduction to IPV6, software defined networks, Open flow.

Module IV (9 Hours)

Link layer and Physical Layer: Introduction to link layer – Error detection (parity, checksum, and CRC), Multiple access protocols (collision and token based), IEEE 802.3 Ethernet, Switching and bridging, Media, Signal strength and interference, Data encoding, Ethernet switches , Routers, MAC, ARP, FIB.

Module V (8 Hours)

IEEE 802.11 Wi-Fi, Bluetooth and cellular networks, Threats and attacks, Network Address Translation, Firewalls, VPNs, Introduction to network management, SNMP, Overview of tools and troubleshooting, Traffic analysis tools , Configuration management.

Text Books

1. Behrouz A Forouzan, Firouz Mosharraf, **"Computer Networks: A top down Approach"**, McGraw Hill Education, 1st Edition (2011).

2. James F Kurose and Keith W Ross, **"Computer Networking: A Top - Down Approach"**, Pearson Education; 6th Edition (2017).

References

1. Kevin R. Fall, W. Richard Stevens, **"TCP/IP Illustrated, Volume 1 -The Protocols"**, Pearson Education, 2nd Edition (2014).

2. Larry Peterson, Bruce Davie, **"Computer Networks, A systems Approach"**, Morgan Kaufmann Publishers, 5th Edition (2011).

3. Uyless Black, **"Computer Networks: Protocols, Standards and Interface"**, Prentice HallIndia Learning Private Limited, 8th Edition (2015).

4. William Stallings, **"Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud"**, Pearson Education, 1st Edition (2016)

5. **The Illustrated Network: How TCP/IP Works in a Modern Network** 2nd edition Walter Goralski Morgan Kaufmann Publications

Course Contents and Lecture Schedule

No	Topic	No of lectures
	Module I	**10 hrs**
1.1	Overview of Computer Networks and the Internet History, Protocols, Review of last mile technologies used for internet access.	2
1.2	Packet switching, Basic ideas about delay, queuing, throughput. Concept of Quality of Service.	2
1.3	Protocol layering. OSI model and TCP model	2
1.4	Application layer protocols - Client-server architecture Network application architecture	2
1.5	Web, HTTP, FTP, SMTP, POP3 and DNS, Peer to-peer file sharing networks	2
	Module II	**10 hrs**
2.1	Transport Layer Protocols: Introduction to transport layer	2
2.2	Multiplexing and demultiplexing, Principles of Reliable data transfer.	2
2.3	Stop-and-wait and Go-back- N design and evaluation	2
2.4	Connection oriented transport TCP, Connection less transport UDP	2
2.5	Principles of congestion control -efficiency and fairness	2
	Module III	**10 hrs**
3.1	Network Layer Protocols: Virtual circuits and datagrams	2
3.2	Principles of routing, internet protocol Ipv4, NAT	2
3.3	Routing algorithms: Link-state and distance vector routing	2
3.4	Routing on the internet RIP, OSPF and BGP, Multicast routing	2
3.5	Introduction to IPV6 and software defined networks	2
	Module IV	**9 hrs**
4.1	Link layer and Physical Layer: Introduction to link layer – Error detection (parity, checksum, and CRC)	2
4.2	Multiple access protocols (collision and token based)	2
4.3	IEEE 802.3, Ethernet, Switching and bridging, Media	2
4.4	Signal strength and interference. Data encoding.	1
4.5	Ethernet switches, Routers MAC, ARP, FIB	2
	Module V	**8 hrs**
5.1	IEEE 802.11 Wi-Fi, Bluetooth, and cellular networks	2
5.2	Threats and attacks, Firewalls, NAT,VPNs	2
5.3	Introduction to network management, SNMP	2
5.4	Overview of tools and troubleshooting, Traffic analysis tools, configuration management.	2

24SJINMCA406	RESEARCH METHODOLOGY	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This course is designed for students to begin research projects which are at the core of MCA course. It aims to give students the tools to identify the research problem, collecting relevant data pertaining to the problem, methodology of carrying out the research and analysis of the result and writing papers.
Prerequisite:	NIL

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the fundamentals concepts of Research Methodology.	K2
CO2	Apply the basic aspects of Research Methodology to formulate a research problem and its plan.	K3
CO3	Deploy numerical/quantitative techniques for data analysis.	K3
CO4	Understand good technical writing.	K2
CO5	Understand the research ethics and publication ethics.	K2

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	1	-	-	1	3
CO2	2	3	3	2	2	1	-	2
CO3	2	3	3	3	1	1	-	2
CO4	2	2	2	1	3	2	1	2
CO5	1	1	1	-	1	1	3	2

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	**40**	**60**	**3 Hours**

Continuous Internal Evaluation Pattern:

Attendance : 8 marks

Continuous Assessment Test (2 numbers) : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Illustrate criteria for good research.
2. Compare types of research.
3. Outline the necessity of defining a problem.

Course Outcome 2 (CO2):

1. Compare and contrast the types of sampling.
2. Summarize the features of good research design.
3. Demonstrate the guidelines for constructing questionnaires.

Course Outcome 3(CO3):

1. Describe some problems in the preparation process.
2. Categorize the limitations of the tests of hypothesis.
3. Define critical value.

Course Outcome 4 (CO4):

1. Categorize types of reports.
2. Recommend the precautions for writing research reports.
3. Describe the significance of report writing.

Course Outcome 5 (CO5):

1. Define ethical values.
2. Illustrate the forms of Publication.
3. Summarize ethical justification.

SYLLABUS

Module 1

Research Methodology - Meaning of research, Objectives of research, Types of Research, Research Approaches, Research Methods versus methodology, Research and scientific method, Research process, Criteria of good research.

Defining research problems - What is a research problem, Selecting the problem, Necessity of defining the problem, Technique involved in defining a problem.

Module 2

Research Design - Meaning of research design, Need for research design, Features of good design, Important concepts relating to research design.

Design of sample surveys - Sample design, Sampling and non-sampling errors, Types of sampling design.

Data Collection - Experiments and Surveys, Collection of primary data, Difference between questionnaire and schedule, Guidelines for constructing questionnaire or schedule, Some other methods of data collection, Collection of secondary data, Selection of appropriate method for data collection.

Module 3

Data preparation - Data preparation process, Some problems in preparation process, Type of analysis.

Testing of Hypothesis - Basic concepts, Testing the hypothesis, Test statistic and critical region, Critical value and decision rule, Procedure for hypothesis testing, Hypothesis testing for mean, Hypothesis testing for proportion, Hypothesis testing for variance, Limitations of the tests of hypothesis.

Module 4

Report Writing - Significance of report writing, Different steps in report writing, Layout of the research report, Types of reports, Mechanics of writing a research report, Precautions for writing research reports.

Module 5

Research Ethics - Ethics, Values, Reason; Value and Value Judgements, Ethics and Ethical Justification, Interests and Consequences.

Doing Research - Understanding rights and responsibilities, defining options, Selecting the Topic, Misconduct.

Publishing - Identifying stakeholders, understanding rights and responsibilities, defining options, Writing, Forms of Publication, Authorship, Peer Review.

Text Books

1. C R Kothari, Gaurav Garg, "Research Methodology - Methods and Techniques" New Age Publishers, Third Edition, 2014
2. Caroline Whitbeck, "Ethics in Engineering Practice and Research" Cambridge University Press, Second Edition, 2012
3. Henriikka Mustajoki, Arto Mustajoki, "A New Approach to Research Ethics", Taylor and Francis Group, First Edition, 2017

Reference Books

1. R Panneerselvam, "Research Methodology", PHI, Second Edition, 2014
2. G C Ramamurthy, "Research Methodology", Dreamtech Press, 2015

Course Contents and Lecture Schedule

No	Topic	No of Lectures
1	**Module 1**	**10 hrs**
	Research Methodology - Meaning of research, Objectives of research, Types of Research, Research Approaches, Research Methods versus methodology, Research and scientific method, Research process, Criteria of good research.	5

	Defining research problems - What is a research problem, Selecting the problem, Necessity of defining the problem, Technique involved in defining a problem.	5
2	**Module 2**	10
	Research Design - Meaning of research design, Need for research design, Features of good design, Important concepts relating to research design.	2
	Design of sample surveys - Sample design, Sampling and non-sampling errors, Types of sampling design.	3
	Data Collection - Experiments and Surveys, Collection of primary data, Difference between questionnaire and schedule, Guidelines for constructing questionnaire or schedule, Some other methods of data collection, Collection of secondary data, Selection of appropriate methodfor data collection.	5
3	**Module 3**	**12 hrs**
	Data preparation - Data preparation process, Some problems in preparation process, type of analysis	2
	Testing of Hypothesis - Basic concepts, Testing the hypothesis, Test statistic and critical region, Critical value and decision rule, Procedure for hypothesis testing, Hypothesis testing for mean, Hypothesis testing for proportion, Hypothesis testing for variance, Limitations of the tests of hypothesis	10
4	**Module 4**	**6 hrs**
	Report Writing - Significance of report writing, Different steps in report Writing.	2
	Layout of the research report, Types of reports, Mechanics of writing a research report, Precautions for writing research reports.	4
5	**Module 5**	**10 hrs**
	Research Ethics - Ethics, Values, Reason; Value and Value Judgements, Ethics and Ethical Justification, Interests and Consequences.	3
	Doing Research - Understanding rights and responsibilities, Defining options, Selecting the Topic, Misconduct.	3
	Publishing - Identifying stakeholders, Understanding rights and responsibilities, Defining options, Writing, Forms of Publication, Authorship, Peer Review.	4

24SJMINCA472	ADVANCED OPERATING SYSTEMS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble:	This course intends to provide insight into more Advanced Operating Systems. Detailed discussion on various concepts like process synchronization, mutual exclusion, resource sharing, concurrency control and security are discussed at algorithm level. Various kinds of advanced operating systems like Distributed Systems, Multiprocessor systems, and Database Systems are included to the level possible within the scope of a single course. More detailed treatment can be done through seminars, assignments and talks by eminent external experts.
Prerequisite:	Basic concepts of desktop computer operating systems.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Identify synchronization problems in operating systems and issues in distributed systems.	K2
CO2	Explain classification of mutual exclusion algorithms and security violations.	K2
CO3	Explain the design of distributed shared memory and issues in load distribution.	K3
CO4	Explain design issues and synchronization in multiprocessor systems.	K3
CO5	Explain synchronization and concurrency control in database systems.	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	1	2					
CO2	3	2	1				1	
CO3	3	1	2					
CO4	3	2	2					
CO5	3	1	1					

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain synchronization using semaphore.
2. Classify Advanced operating systems.
3. Illustrate limitation of Lamports clocks.

Course Outcome 2 (CO2):

1. Explain some of the algorithms for mutual exclusion.
2. Explain potential security violations.
3. Compare the Lamport’s algorithm and Rickart-Agarwala algorithm.

Course Outcome 3(CO3):

1. Explain major design issues and building mechanisms in Distributed file systems.
2. Explain important algorithms for implementing DSM.
3. Explain issues in load distribution.

Course Outcome 4 (CO4):

1. Explain system architecture of Multiprocessor systems.
2. Explain design issues in Database Multiprocessor Systems.
3. Explain how virtualization is implemented.

Course Outcome 5 (CO5):

1. Explain Lock based algorithms for concurrency control in Database Systems.
2. Illustrate Timestamp based algorithms for concurrency control in Database Systems.
3. Explain design issues in Database Systems.

Syllabus

Module	Contents	Hours
I	**Overview**: Functions of Operating System –Design Approaches –Types of Advanced Operating Systems. **Synchronization Mechanisms**: Concept of Processes and Threads –The Critical Section Problem – Other Synchronization Problems:– Monitor –Serializer – Path Expressions. **Distributed Operating Systems**:- Issues in Distributed Operating System – Communication Networks and Primitives –Lamport's Logical clocks – Causal Ordering ofMessages.	10
II	**Distributed Mutual Exclusion**:- Classification - Requirements – Measuring Performance – Lamport's Algorithm – Rickart-Agarwala Algorithm – Suzuki- Kasami's Broadcast Algorithm. **Security:-**Potential Security Violations – Design Principles for Secure Systems –The Access Matrix Model and Implementation- The Access Control list Method.	10
III	**Distributed Resource Management**: Mechanisms for building Distributed File Systems – Design Issues - Distributed Shared Memory – Algorithms for ImplementingDistributed Shared memory – Issues in Load Distributing – Components of Load Distributing Algorithm – Sender- Initiated Algorithm – Receiver- Initiated Algorithm.	10
IV	**Multiprocessor Operating Systems**: Basic Multiprocessor System Architectures – Interconnection Networks –Structures – Design Issues – Threads – Process - Synchronization – Processor Scheduling – Memory Management – Virtualization – Types of Hypervisors – Paravirtualization – Memory Virtualization – I/O Virtualization.	8
V	**Database Systems**: Problem of Concurrency Control – Serializability – Basic Synchronization Primitives for Concurrency Control – Lock-Based Algorithms – Time-Stamp Based Algorithms – Optimistic Algorithms.	10

Textbooks:

1. Mukesh Singhal and Niranjan G. Shivaratri, "**Advanced Concepts in Operating Systems**– Distributed, Database, and Multiprocessor Operating Systems",Tata McGraw-Hill,2001.

2.Andrew S. Tanenbaum, "**Modern Operating Systems**", 3rd Edition, Prentice Hall,2012.

Reference Books:

1. Pradeep K Sinha, "**Distributed Operating Systems: Concepts and Design**", PrenticeHall of India, 2007.
2. George Coulouris, Jean Dollimore,Tim Kindberg, Gordon Blair, **"DistributedSystems,Concepts and Design"**, 5th Edtn,Pearson, 2019

Web Resources

1. https://www.my-mooc.com/en/mooc/advanced-operating-systems--ud189/
2. https://www.slideshare.net/slideshow/cs9222-advanced-operating-system/54590046

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction**	
1.1	**Overview**: Functions of Operating System –Design Approaches –Types of Advanced Operating Systems.	2
1.2	**Synchronization Mechanisms**: Concept of Processes and Threads – The Critical Section Problem – Other Synchronization Problems:– Monitor –Serializer – Path Expressions.	4
1.3	**Distributed Operating Systems**:- Issues in Distributed Operating System – Communication Networks And Primitives –Lamport's Logical clocks – Causal Ordering of Messages.	4
2		
2.1	**Distributed Mutual Exclusion**:- Classification - Requirements – Measuring Performance – Lamport's Algorithm.	2
2.2	Rickart-Agarwala Algorithm – Suzuki- Kasami's Broadcast Algorithm.	3
2.3	**Security :**Potential Security Violations – Design Principles for Secure Systems –The Access Matrix Model and Implementation- The Access Control list Method.	5
3		
3.1	**Distributed Resource Management**: Mechanisms for building Distributed File Systems – Design Issues.	3
3.2	**Distributed Shared Memory** – Algorithms for Implementing Distributed Shared memory.	3
3.3	**Load Distribution** : Issues in Load Distributing – Components of Load Distributing Algorithm – Sender- Initiated Algorithm – Receiver- Initiated Algorithm.	4

4		
4.1	**Multiprocessor Operating Systems**: Basic Multiprocessor System Architectures – Interconnection Networks – Structures.	3
4.2	Design Issues – Threads – Process Synchronization - Processor Scheduling – Memory Management	3
4.3	Virtualization – Types of Hypervisors – Paravirtualization – Memory Virtualization – I/O Virtualization.	2
5		
5.1	**Database Systems**: Problem of Concurrency Control – Serializability – Basic Synchronization Primitives for Concurrency Control – Lock-Based Algorithms.	5
5.2	Time-Stamp Based Algorithms.	3
5.3	Optimistic Algorithms.	2

24SJINMCA482	BUSINESS MANAGEMENT	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	-	4

Preamble:	The primary aim of this course is to understand basic principles of management and accounting. In our day-to-day life managers will have to manage so many resources in the present-day complex business environment. By effective and efficient management, the goals of the organisation can be attained. This course is intended to give an idea regarding managing the resources for the effective performance of the organisation and decision-making in everyday life. Basic idea regarding bookkeeping and accounting is also required for managers for taking decisions.
Prerequisite:	NIL

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand management as a process.	K2
CO2	Critically analyse and evaluate management theories and practices	K2
CO3	Perform planning and organising for an organisation	K3
CO4	Do staffing and related human resource development function	K3
CO5	Take proper decisions to get a competitive advantage and understand basic concepts in book keeping and accounting	K2

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3				2			2
CO2	3	3			2			1
CO3	3		3		2			
CO4	3				3			
CO5	2				3	2		2

Mark distribution

Total Marks	C IE	ES E	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks

Continuous Assessment Test (2 numbers) : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment

Course Outcome C01: Describe various functions of management.

Course Outcome CO2 : Explain different theories of management thought.

Course Outcome CO3 : Illustrate different steps in planning.

Course Outcome CO4 : Describe different types of training methods for employees in an organisation.

Course Outcome CO5 : Explain the decision process in an organisation and the procedure of preparation of balance sheet with case example.

SYLLABUS

Module I

Introduction to Management: Basic Managerial Concepts, Levels of management, Managerial Skills, Managerial role. Management functions- Planning, Organising, Staffing, leading and Controlling. Early Contributions in Management: Management thought - Classical approach, scientific management, contributions of Taylor, Gilbreths,

Fayol's 14 principles of management. Human relation approach - contribution of Elton Mayo Systems approach - organization as an open system and Contingency approach.

Module II

Planning: Nature and importance of planning, types of plans - Steps in planning, Levels of planning - The Planning Process - MBO definition and process, SWOT Analysis, importance. Organising : Nature of organizing,-span of control in management, factors affecting span of control- Authority and responsibility.

Organisation structure - Formal and informal, Types of organization structure line, line and staff, functional, divisional, project, matrix, virtual form of organisations.

Module III

Staffing and related HRD Functions: meaning, nature, staffing process, Job analysis and manpower planning, job description and job specification, Recruitment & selection, selection process, Tests and interviews. Training and development - concept and methods ,Performance appraisal- concept and methods.

Module IV

Managerial Decision Making and controlling : Decision making –types of decisioins, decision making process, Decision Making Tools, Importance of controlling, Techniques of controllingBreak Even Analysis, Budgetary Control - Benchmarking –importance and limitations of benchmarking, Six Sigma importance, limitations and process of six sigma, Total Quality Management- Introduction to marketing management-Marketing mix- product life cycle.

Module V

Book- Keeping and Accountancy -Elements of Double Entry -Book- Keeping - rules for journalizing -Ledger accounts –Cash book- – Trial Balance- Method of Balancing accounts-the journal proper (simple problems). Final accounts: Preparation of trading and profit and loss Account- Balance sheet (with simple problems) - Introduction to Accounting packages (Description only).

References

1. L M Prasad, “Principles of Management”, Sultan Chand & Sons, 8th Edition (2010)

2. Peter F Drucker, “The Practice of Management”, Butterworth-Heinemann publication, 2nd Edition (2007)

3. Harold Koontz and Heinz Weihrich, “Essentials of Management”, McGraw Hill Education, 10th Edition (2015).

4. Robbins and Coulter, Management, Pearson Education 13th Edition, 2016,

5. R N Gupta, “Principles of Management”, S. Chand & Company Ltd., (2010)

6. Tripathi, “Principles of Management”, McGraw Hill Education, 5th Edition (2012)

7. Double Entry book Keeping – Batliboi 8. A Systematic approach to Accounting: Dr K.G. Chandrasekharan Nair

Suggested MOOCs

1. Management Functions https://nptel.ac.in/courses/122108038

2. Leadership https://onlinecourses.nptel.ac.in/noc23_hs143/preview

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	Introduction to Management: Basic Managerial concepts	2 Hours
1.1	Levels of management, Managerial Skills.	2 Hours
1.2	Management roles	1 Hour
1.3	Management functions	2 Hours
1.4	Early Contributions in Management: Management thought - Classical approach, scientific management, contributions of Taylor, Gilbreths, Fayol's 14 principles of management. Human relation approach - contribution of Elton Mayo Systems.	3 Hours
2	Planning: Nature and importance of planning, types of plans - Steps in planning, Levels of planning - The Planning Process	2 Hours
2.1	MBO definition and process, SWOT Analysis, importance. .	2 Hours
2.2	Organising : Nature of organizing,-span of control in management, factors affecting span of control- authority and responsibility.	3 Hours
2.3	Organisation structure - Formal and informal, Types of organization structure line, line and staff, functional, divisional, project, matrix, virtual form of organisations	2 Hours
3	Staffing and related HRD Functions: meaning, nature, staffing process	2 Hours
3.1	Job analysis and manpower planning, job description and job specification	2 Hours
3.2	Recruitment & selection, selection process, Tests and interviews. Training and development - concept and methods	3 Hours
3.3	Performance appraisal - concept and methods.	2 Hours

4	Managerial Decision Making and controlling : Decision making – types of decisions, decision making process, Decision Making Tools.	2 Hours
4.1	Importance of controlling, Techniques of controlling- Break Even Analysis, Budgetary Control.	2 Hours
4.2	Benchmarking –importance and limitations of benchmarking	2 Hours
4.3	Six Sigma importance, limitations and process of six sigma	2 Hours
4.4	Total Quality Management.	2 Hours
4.5	Introduction to marketing management-Marketing mix- product life cycle.	2 Hours
5	Book- Keeping and Accountancy -Elements of Double Entry -Book Keeping.	2 Hours
5.1	Rules for journalizing -Ledger accounts –Cash book.	3 Hours
5.2	Trial Balance- Method of Balancing accounts- (simple problems). Final accounts: Preparation of trading and profit and loss Account- Balance sheet (with simple problems).	3 Hours
5.3	Introduction to Accounting packages.	2 Hours

24SJINMCA492	IPR AND CYBER LAWS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble:	This course intends to provide insight into Intellectual Property Rights and Cyber Laws. It includes detailed discussion on various intellectual property rights, procedures to apply for copyrights &patents, legalities of intellectual property to avoid plagiarism and other IPR related crimes. Effectiveness of cyber-laws and other countermeasures against cybercrime and cyber warfare are discussed in detail. Various kinds of Intellectual Property issues in cyberspace and the growth and development of the law in this regard are included to the level possible within the scope of a single course. More detailed treatment can be done through seminars, assignments and talks by eminent external experts including industry.
Prerequisite:	General awareness on internet essentials, web technologies, e-commerce.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Explain the fundamentals of IPR and patents.	K2
CO2	Apply intellectual property related tools such as trademark and copyright to real problems.	K3
CO3	Discuss Industrial designs, trade secret and geographic Indications.	K2
CO4	Describe laws governing cyberspace and analyze the role of Internet Governance in framing policies for Internet security.	K2
CO5	Discuss different types of cybercrimes and penalties under IT Act.	K2

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2			1			2	
CO2	3	1		1			1	
CO3	3			1			1	
CO4	3	1		1			1	
CO5	3	1					1	1

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Discuss the need for protection of intellectual property.
2. Explain TRIPS Agreement.
3. Illustrate types of patent applications.

Course Outcome 2 (CO2)

1. Explain Trademark Infringement and Protection of trademarks.
2. Explain the rights conferred by copyright, registration and ownerships of copyrights.
3. Discuss about software copyright.

Course Outcome 3(CO3):

1. Discuss the need for protection of design and explain Design Act, 2000.
2. Explain basic concepts of Geographic Indications such as filing, granting and Protection of geographic indications.
3. Describe the procedure of discovering and protecting of trade secret.

Course Outcome 4 (CO4):

1. Explain the need for cyber laws.
2. Discuss protection of copyright on cyberspace.
3. Explain ISP in cyberspace.

Course Outcome 5 (CO5):

1. Explain different amendments on IT Act 2000.
2. Discuss Terrorism on cyberspace.
3. Explain offences of misrepresentation.

Syllabus

Module	Contents	Hours
I	Fundamentals of IPR- Introduction – Intellectual property – Need for protection of intellectual property – WIPO – Intellectual property rights and development – Rationale of protection – TRIPS Agreement **Patents :** – Introduction – Patentable and Non-patentable Invention – Types of patent applications – Guidelines for registration of patent – patent filing – grant of patent – types of patent documents.	10
II	**Trademarks**– Introduction – Guidelines for registration- Requirements for filing trademarks – Trademark Infringement – Protection of trademarks – **Copyright** – Introduction – Rights conferred by copyright – registration – ownerships – terms – transfer of copyrights – copyright infringement – databases and copyright- **Software Copyright**– Introduction – Need of software copyright – classification of software according to copyright – software auditing –copyright notice – transfer of copyright.	10
III	**Industrial Designs**– Introduction – Need for protection of design – requirements for registration of designs – Design Act,2000 – Duration of registration of design – application procedure – **Geographic Indications** –Introduction – Filing – **Granting** – Protection of geographic indications. **Trade Secret** – definition – discovering and protecting of trade secret.	10
IV	**Cyber law** - Need for cyber laws - Historical perspective - cyberspace - deception by squatting in cyberspace - protection of copyright on cyberspace - infringement of copyright on cyberspace - linking, hyperlinking and framing - ISP in cyberspace - cyberspace and protection of patents in India.	8
V	**Information Technology Act and Punishments**- Introduction to IT Act 2000- Amendments on IT Act - Violation of the right of privacy in cyberspace/internet-punishment for violation of privacy, breach of confidentiality and privacy under IT act-Terrorism on cyberspace Overview of cybercrimes-offences by intermediaries- offences related to protected system- offences of misrepresentation- punishment for Abetment and Attempt to commit offences under the IT act.	10

Textbooks:

1. Dr. R. Radhakrishnan and Dr. S. Balasubramanian, "**Intellectual Property Rights: Text and Cases**", Excel Books
2. Harish Chander, "**Cyber Law and IT Protection**", PHI Learning Pvt.Ltd.

Reference Books:

1. D.Bainbridge, "**Introduction to Computer Law**", Pearson Education
2. RohasNagpal, "**Cyber Crime & Corporate Liability**", CCH, 2008

Web Resources

1. https://www.udemy.com/course/cyber-security-law/
2. https://www.coursera.org/specializations/introduction-intellectual-property

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	
1.1	**Fundamentals of IPR-** Introduction – Intellectual property – Need for protection of intellectual property	2
1.2	WIPO – Intellectual property rights and development – Rationale of protection – TRIPS Agreement	3
1.3	**Patents :** – Introduction – Patentable and Non-patentable Invention – Types of patent applications – Guidelines for registration of patent – patent filing – grant of patent – types of patent documents	5
2	**Module 2**	
2.1	**Trademarks**– Introduction – Guidelines for registration – Requirements for filing trademarks – Trademark Infringement – Protection of trademarks	3
2.2	**Copyright** – Introduction – Rights conferred by copyright – registration – ownerships – terms – transfer of copyrights – copyright infringement – databases and copyright	3
2.3	**Software Copyright**– Introduction – Need of software copyright – classification of software according to copyright – software auditing – copyright notice – transfer of copyright.	4
3	**Module 3**	

3.1	**Industrial Designs**– Introduction – Need for protection of design – requirements for registration of designs – Design Act,2000 – Duration of registration of design – application procedure	4
3.2	**Geographic Indications** – Introduction – Filing **Granting** – Protection of geographic indications**.**	4
3.3	**Trade Secret –** definition – discovering and protecting of trade secret.	2
4	**Module 4**	
4.1	**Cyber law** - Need for cyber laws - Historical perspective - cyberspace - deception by squatting in cyberspace.	3
4.2	Protection of copyright on cyberspace - infringement of copyright on cyberspace - linking, hyper linking and framing -	3
4.3	ISP in cyberspace - cyberspace and protection of patents in India.	2
5	**Module 5**	
5.1	**Information Technology Act and Punishments**- Introduction to IT Act2000- Amendments on IT Act	2
5.2	Violation of the right of privacy in cyberspace/internet-punishment for violation of privacy, breach of confidentiality and privacy under IT act- Terrorism on cyberspace overview of cybercrimes	4
5.3	Offences by intermediaries- offences related to protected system- offences of misrepresentation-punishment for Abetment and Attempt to commit offences under the IT act.	4

24SJINMCA432	ADVANCED DBMS LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble:	This course is to provide understanding on relational and non-relational database systems and its design. The course covers SQL, PL/SQL and NoSQL programs which are essential for the development and deployment of web-based applications. Also, this course serves as a prerequisite for many advanced courses in Data Science areas.
Prerequisite:	Database Management Systems.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO 1	Design and build a simple relational database system and demonstrate competence with the fundamentals tasks involved with modelling, designing and implementing a database.	K5
CO 2	Apply PL/SQL for processing databases.	K5
CO 3	Comparison between relational and non-relational (NoSQL) databases and the configuration of NoSQL Databases.	K5
CO 4	Apply CRUD operations and retrieve data in a NoSQL environment.	K5
CO 5	Understand the basic storage architecture of distributed file systems.	K5
CO 6	Design and deployment of NoSQL databases with real time requirements.	K5

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	1	1	2
CO2	3	1	2	3	1	1	1	2
CO3	2	3	2	3	1	1	1	1
CO4	3	1	2	3	2	1	1	1
CO5	3	2	1	2	1	1	2	1
CO6	2	3	3	3	2	2	1	3

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

<table>
<tr><td colspan="4">Maximum Marks: 50</td></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 marks</td></tr>
<tr><td rowspan="4">Problem solving

(Based on difficulty level, one or more questions may be given)</td><td>Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15 %</td></tr>
<tr><td>Formatted output and Pushing to remote Git repository</td><td>20%</td></tr>
<tr><td colspan="3">Total Marks</td><td>50 marks</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Creation of a database using DDL commands including integrity constraints.

2. Create an application to apply Data Manipulation Language (DML) commands to modify the database.
3. Apply DCL and TCL commands to impose restrictions on databases.
4. Create an application to retrieve data from databases using select, views.
5. Create an application to use joins for query optimization.

Course Outcome 2 (CO2)

1. Construct PL/SQL code for sample databases.

Course Outcome 3(CO3):

1. Compare relational and non-relational databases.
2. Understand the installation and configuration of NoSQL Databases.

Course Outcome 4 (CO4):

1. Build sample collections/documents to perform query operations.

Course Outcome 5 (CO5):

1. Build sample collections/documents to perform the shell commands like replica set, indexing etc.

Course Outcome 6(CO6):

1. Develop sample applications using any of the front-end tools and NoSQL.
2. Usage of concerned Online/Cloud Storage Management Systems like MongoDB Atlas, Cassandra DataStax etc.
3. Deployment of NoSQL in Cloud: Google Bigtable/ Amazon DynamoDB/ Azure Cosmos DB.

SYLLABUS

1. An overview of relational database design using MySQL/ MariaDB/ PostgreSQL etc. (Apply the following basic queries on an Employee/ Student database etc.)
   a. DDL Commands
   b. DML Commands
   c. Imposing restrictions on database (DCL & TCL Commands)
   d. Accessing database (SELECT, Filtering using WHERE, HAVING, GROUP BY, ORDER BY Clauses, Subquery and View)
   e. Optimizing databases (Join, Aggregate & Set operations, Other operators

like arithmetic, logical, special etc.)

2. PL/SQL Programs (Trigger, Cursor, Stored Procedures and Functions)
3. Introduction to NoSQL Databases.
    a. Installation and configuration of any one of the NoSQL databases - MongoDB/ Cassandra/ HBase/ CouchDB/ Amazon DynamoDB/ Redis/ Neo4j etc.
4. Designing Databases using NoSQL
5. Query Processing
    a. Performing CRUD operations
    b. Retrieving Data from a NoSQL database
    c. Usage of aggregate functions, regular expressions etc.
6. NoSQL Administration
    a. Security, Monitoring & Backup
    b. Create Users and Roles
7. NoSQL shell commands
    a. Perform Sharding, Replication (Master-Slave/ Master-Less/ Peer-to-Peer Architectures), Clustering, Partitioning, Indexing (Corresponding to the selected NoSQL Database)
8. Deployment
    a. Local Deployment
        i. NoSQL and Front-End: PHP/Java/Python (MongoDB/ Cassandra etc.)
    b. Cloud Deployment
        i. NoSQL and Cloud: Amazon DynamoDB/ Google Bigtable/ Azure Cosmos DB
        ii. Familiarization of Atlas/ DataStax corresponding to the selected NoSQL Database

Text Books

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan,” Database System Concepts”, McGraw Hill Education, 6th Edition (2011)

2. Guy Harrison, “Next Generation Databases: NoSQL, NewSQL, and Big Data”, Apress, 1st Edition (14 December 2015)

Reference Books

1. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", McGraw Hill, 3rd Edition (2014).
2. HBase: The Definitive Guide. Lars George O'Reilly Media; August 2011, ISBN: 9781449315771
3. Shashank Tiwari. Professional NoSQL. John Wiley and Sons. ISBN: 978-0-470-94224-6.
4. MongoDB Administrator's Guide, Cyrus Dasadia, October 2017, Packet Publishing ISBN: 9781787126480
5. Cassandra: The Definitive Guide Distributed Data at Web Scale, 1st Edition, Eben Hewitt, Jeff Carpenter, O'Reilly Media; November 2010

Web Resources

1. Database Management System https://onlinecourses.nptel.ac.in/noc22_cs91/preview
2. Databases: SQL https://www.edx.org/course/databases-5-sql
3. Introduction to MongoDB https://www.coursera.org/learn/introduction-to-mongodb
4. Apache Cassandra https://www.edureka.co/cassandra
5. NoSQL systems https://www.coursera.org/learn/nosql-databases
6. https://hbase.apache.org/
7. https://aws.amazon.com/dynamodb/

Course Contents and Lecture Schedule

Sl No	Topic	No. of hours
1	An overview of relational database design using MySQL/ MariaDB/ PostgreSQL etc.(Apply the following basic queries on an Employee/ Student database etc.)	6
1.1	DDL Commands DML Commands Imposing restrictions on database (DCL & TCL Commands)	3
1.2	Accessing database (SELECT, Filtering using WHERE, HAVING, GROUP BY, ORDER BY Clauses, Subquery and View)	3

	Optimizing databases (Join, Aggregate & Set operations, Other operators like arithmetic, logical, special etc.)	
2	PL/SQL Programs.	4
2.1	Trigger, Cursor, Stored Procedures and Functions.	4
3	Introduction to NoSQL Databases.	2
3.1	Installation and configuration of any one of the NoSQL databases - MongoDB/ Cassandra/ HBase/ CouchDB/ Amazon DynamoDB/ Redis/ Neo4j etc.	2
4	Designing Databases using NoSQL	2
5	Query Processing	8
5.1	Performing CRUD operations Retrieving Data from a NoSQL database Usage of aggregate functions, regular expressions etc.	8
6	NoSQL Administration	2
6.1	Security, Monitoring & Backup Create Users and Roles	2
7	NoSQL shell commands	6
7.1	Perform Sharding, Replication (Master-Slave/ Master-Less/ Peer-to-Peer Architectures), Clustering, Partitioning, Indexing (Corresponding to the selected NoSQL Database	6
8	Deployment	16
8.1	Local Deployment NoSQL and Front-End: PHP/Java/Python (MongoDB/ Cassandra etc.)	4
8.2	Cloud Deployment NoSQL and Cloud: Amazon DynamoDB/ Google Bigtable/Azure Cosmos DB	8
8.3	Familiarization of Atlas/ DataStax corresponding to the selected NoSQL Database	4
9	Micro project	10

24SJINMCA434	NETWORKING & SYSTEM ADMINISTRATION LAB	CATEGORY	L	T	P	CREDIT
		GENERAL	0	1	3	1

Preamble:	The syllabus is prepared for equipping MCA Graduates to design, troubleshoot, model and evaluate computer networks. The students can explore through network simulation programs and thereby evaluate their design approaches and expected network performance outcome. The students can achieve a thorough understanding on network program execution using Python language, best suited for Network Administration under various operating systems.
Prerequisite:	24SJINMCA404 Advanced Computer Networks

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Design a computer network and configured by relevant protocols.	K3
CO2	Design socket programming for client server interaction	K5
CO3	Create a simple Pyro Server, XML-RPC server and to know how to connect to SSH server and remotely execute a command	K3
CO4	Experiment an MD5 checksum on files and in removal of duplicate files.	K5
CO5	Experiment with SNMP network discovery and their relationships, their inter-dependencies	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	3	2	1	3	2	2	2
CO2	2	2	3	3	3			1
CO3	3	3	3	2	3			
CO4	3	3	3	3	3			
CO5	3	3	3	1	2			2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern

Attendance : 8 marks

Continuous Assessment Test (2 numbers) : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern:

Lab exam will be conducted by internal examiner.

Course Level Assessment

Course Outcome 1 (CO1):

1. Explore basic network commands.
2. Demonstrate a network using Distance vector routing protocol.

Course Outcome 2 (CO2):

1. Design a client server application program in Python using UDP.
2. Design a client server application program in Python for two – way chat using TCP.

Course Outcome 3 (CO3):

1. Demonstrate TCP port checker in Python.
2. Demonstrate socket-based web server checker.

Course Outcome 4 (CO4):

1. Convince through the use of MD5 checksum on files.

Course Outcome 5 (CO5):

1. Demonstrate procedure for data center discovery.
2. Apply methods for net-SNMP and configuring SNMP agents on Linux server.

Reference Books

1. Python for Unix and Linux System Administration by Noah Gift, Jeremy M. Jones

Web Resources

1. https://www.coursera.org/learn/cloud-computing?specialization=cloud-computing
2. https://www.coursera.org/learn/cloud-computing-2?specialization=cloud-computing
3. https://alison.com/course/introduction-to-server-administration-services-revised

MOOC Courses

1. Computer Networks and Internet Protocol https://nptel.ac.in/courses/106105183/
2. Introduction on Computer Networks https://nptel.ac.in/courses/106106091/

List of Lab Experiments

1. Execute basic network commands ipconfig, ping, traceroute, nslookup etc.
2. Apply command pathping.
3. Design a network topology using packet tracer software.
4. Design a network using Distance vector routing protocol.
5. Design a network using Link State routing protocol
6. Write a program for implementing an echo server in Python.
7. Write a client server application program in Python using UDP.
8. Write a client server application program in Python for two – way chat using TCP.
9. Implement TCP port checker in Python.
10. Implement socket-based web server checker.
11. Implement httplib-based web server checker
12. Implement a simple Pyro server.
13. Implement using shutil module to copy a data tree.
14. Evaluate MD5 checksum on a directory tree to find duplicates.
15. Implement basic Net-SNMP session module.
16. Write SNMP configuration file to display 'Hello World'.

Course Contents and Lecture Schedule

No.	Topic	No. of Hours
1	**Module 1**	**12 Hrs.**
1.1	Study of Networking- a. Study of network devices in detail b. Study of network IP c. Connect the computers in local area network d. Study of basic network command and Network configuration commands. e. Configure a Network topology using packet tracer software. f. Configure a network using Distance vector routing protocol. g. Configure a network using Link state routing protocol.	12

2	**Module 2**	**8 Hrs.**
2.1	System Administration using Python programming a. Socket b. TCP Client-server communication program using socket c. UDP Client-server communication program using socket. d. Echo server implementation.	8
3	**Module 3**	**8 Hrs.**
3.1	a. TCP port checker b. Socket based web server checker	2
3.2	a. Httplib-based web server checker	2
3.3	a. Simple XML-RPC server b. Simple Pyro server c. Connecting to an SSH server and remotely executing a command	4
4	**Module 4**	**12 Hrs.**
4.1	a. Using the shutil module to copy a data tree b. Moving a data tree with shutil.	4
4.2	a. Performing an MD5 checksum on files. b. Performing an MD5 checksum on a directory tree to find duplicates	6
4.3	a. Using IPython and Net-SNMP with Python bindings b. Basic Net-SNMP session module	2
5	**Module 5**	**8 Hrs.**
5.1	a. Basic data center discovery b. SNMP configuration file with Hello World	8

ST.JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

MCA Integrated

SEMESTER 9

24SJINMCA501	DATA SCIENCE & MACHINE LEARNING	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	This is an introductory course on data science and basic concepts behind various machine learning techniques. Machine learning is the study of adaptive computational systems that improve their performance with experience. At the end of the course the students should be able to design and implement machine learning solutions to classification, regression, and clustering problems and to evaluate and interpret the results of the algorithms.
Prerequisite:	Probability and Statistics, Linear Algebra, Programming in Python/R.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Discuss the fundamental concepts of data science and data visualization techniques.	Understand (K2)
CO2	Apply the basics of machine learning, lazy learning and probabilistic learning algorithms to solve data science problems.	Apply (K3)
CO3	Describe decision trees, classification rules & regression methods and how these algorithms can be applied to solve data science problems.	Apply (K3)
CO4	Solve data science problems using neural networks and support vector machines.	Apply (K3)
CO5	Discuss clustering using k-means algorithm and evaluate & improve the performance of machine learning classification models.	Understand (K2)

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	1	1	-	-	-	-	-	3
CO2	3	3	3	-	-	-	-	3
CO3	3	3	3	-	-	-	-	3

CO4	3	3	3	-	-	-	-	3
CO5	3	3	3	-	-	-	-	3

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks

Continuous Assessment Test (2 numbers) : 20 Marks

Assignment/Quiz/Course project : 12 Marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. What is data science and why do we need data science?
2. Explain the data science classification and illustrate data science tasks.
3. Describe the various methods to understand data.
4. Explain the typical methods to visualize data.

Course Outcome 2 (CO2)

1. Explain the differences between supervised and unsupervised machine learning algorithms.
2. Describe the key concepts that define nearest neighbour classifiers, and why they are considered "lazy" learners.
3. Explain how to apply k-NN classifier in a data science problem.
4. State Bayes' theorem in statistics. Outline the Naive Bayes algorithm to build classification models.
5. Use Naive Bayes algorithm to determine whether a red domestic SUV car is a stolen car or not using the following data:

Example	Color	Type	Origin	Stolen?
1	red	sports	domestic	yes
2	red	sports	domestic	no
3	red	sports	domestic	yes
4	yellow	sports	domestic	no
5	yellow	sports	imported	yes
6	yellow	SUV	imported	no
7	yellow	SUV	imported	yes
8	yellow	SUV	domestic	no
9	red	SUV	imported	no
10	red	sports	imported	yes

Course Outcome 3 (CO3):

1. Classify data science tasks using decision trees and classification rule learners.
2. Discuss the various feature selection measures.
3. How to simplify a decision tree by pruning.
4. Describe how to construct classification rules from decision trees.
5. Explain the concepts of regression and correlation.
6. How to estimate a linear regression model.
7. Consider the following set of training examples:

Instance	Classification	a1	a2
1	+	T	T
2	+	T	T

3	-	T	F
4	+	F	F
5	-	F	T
6	-	F	T

a) Find the entropy of this collection of training examples with respect to the target function “classification”?

b) Calculate the information gain of a2 relative to these training examples?

Course Outcome 4 (CO4):

1. Explain how artificial neural networks mimic human brain to model arbitrary functions and how these can be applied to real-world problems.
2. Describe different activation functions and network topology.
3. Discuss basic idea behind the backpropagation algorithm.
4. Explain how a support vector machine can be used for classification of linearly separable data.
5. How to compute the distance of a point from a hyperplane.
6. How the kernel trick is used to construct classifiers in nonlinearly separated data.

Course Outcome 5 (CO5):

1. Explain how the clustering tasks differ from the classification tasks.
2. How clustering defines a group, and how such groups are identified by k-means clustering algorithm.
3. Find the three clusters after one epoch for the following eight examples using the k- means algorithm and Euclidean distance: A1 = (2,10), A2 = (2,5), A3 = (8,4), A4 = (5,8), A5 = (7,5), A6 = (6,4), A7 = (1,2), A8 = (4,9). Suppose that the initial seeds (centres of each cluster) are A1, A4 and A7.
4. Explain the various matrices used to measure the performance of classification algorithms
5. Explain the concepts of bagging and boosting.
6. Suppose 10000 patients get tested for flu; out of them, 9000 are actually healthy and 1000 are actually sick. For the sick people, a test was positive for 620 and negative for 380. For the healthy people, the same test was positive for 180 and

negative for 8820. Construct a confusion matrix for the data and compute the precision and recall for the data.

SYLLABUS

MODULE	CONTENTS	HOURS
I	**Introduction to data science:** Data science classification, Data science process – Prior knowledge, Data preparation, Modelling, Application. **Data exploration:** Data sets, Descriptive statistics for univariate and multivariate data. **Data visualisation:** Histogram, Quartile plot, Distribution chart, Scatter plot, Bubble chart, Density chart	9
II	**Introduction to machine learning:** How machines learn - Data storage, Abstraction, Generalisation, Evaluation, Machine learning in practice - Types of machine learning algorithms. **Lazy learning:** Classification using K-Nearest Neighbour algorithm - Measuring similarity with distance, Choice of k, Preparing data for use with k-NN. **Probabilistic learning:** Understanding Naive Bayes - Conditional probability and Bayes theorem, Naive Bayes algorithm for classification, The Laplace estimator, Using numeric features with Naive Bayes.	10
III	**Decision tree learning:** Concept of decision tree, Divide and conquer approach, C5.0 Decision tree algorithm, Choosing the best split, Pruning the decision tree. **Classification rules learning:** Concept of classification rules, Separate and conquer approach, The 1R algorithm, Rules from decision trees. **Regression methods:** Concept of regression, Simple linear regression, Ordinary least squares estimation, Correlations, Multiple linear regression.	10
IV	**Neural Network Learning:** Artificial neurons, Activation functions, Network topology, Training neural networks with backpropagation.	10

	Support Vector Machines: Hyperplanes, Classification using hyperplanes, Maximum margin hyperplanes in linearly separable data, Using kernels for non-linear spaces.	
V	**Clustering:** The k-means clustering algorithm, Using distance to assign and update clusters, Choosing number of clusters. **Evaluating model performance:** Confusion matrices, Precision and recall, Sensitivity and specificity, Precision and recall, F-measure, ROC curves, Cross validation - K-fold cross validation, Bootstrap sampling. **Improving model performance:** Bagging, Boosting, Random forests.	8

Text Books

1. Vijay Kotu, Bala Deshpande, Data Science Concepts and Practice, Morgan Kaufmann Publishers 2018 (Module 1)
2. Brett Lantz, Machine Learning with R, second edition, PackT publishing 2015 (Modules 2 to 5)

Reference Books

1.Michael Steinbach, Pang-Ning Tan, and Vipin Kumar, Introduction to Data Mining, Pearson 2016.

2. Jiawei Han, Micheline Kamber and Jian Pei, Data mining Concepts and techniques Morgan Kaufmann Publishers 2012

3. Peter Harrington, Machine Learning in action, Dreamtech publishers 2012

4. Dr M Gopal, Applied Machine learning, McGraw Hill Education Private Limited

5. E. Alpayidin, Introduction to Machine Learning, Prentice Hall of India (2005)

6.T. Hastie, RT Ibrashiran and J. Friedman, The Elements of Statistical Learning, Springer 2001

7.Data Science from Scratch: First Principles with Python, Joel Grus, O'Reilly, First edition, 2015

8.Introducing Data Science, Davy Cielen, Arno D. B. Meysman, Mohamed Ali, Manning Publications Co., 1st edition, 2016

Web Resources:

1. https://www.coursera.org/learn/machine-learning
2. https://www.coursera.org/learn/data-scientists-tools

Course Contents and Lecture Schedule:

No	Topic	No: of Lectures (Hours)
1	**Module 1**	
1.1	Introduction to data science - What is data science? Why data science?	2
1.2	Data science classification	1
1.3	Data science process - Prior knowledge, Data preparation, Modelling, Application	2
1.4	Data exploration- Data sets, Descriptive statistics for univariate and multivariate data	2
1.5	Data visualization – Histogram, Quartile plot, Distribution chart, Scatter plot, Bubble chart, Density chart	2
2	**Module 2**	
2.1	How machines learn – Data storage – Abstraction Generalisation – Evaluation	1
2.2	Machine learning in practice – Types of machine learning algorithms	1
2.3	Classification: Lazy learning - K-Nearest Neighbour algorithm	2
2.4	Measure of similarity, Choice of k	1
2.5	Preparing data for use with k-NN	1
2.6	Probabilistic Learning: Conditional probability and Bayes theorem.	2
2.7	Naive Bayes algorithm 2	2
3	**Module 3**	
3.1	Concept of decision tree, Divide and conquer approach	1
3.2	C5.0 Decision tree algorithm	1
3.3	Choosing the best split, Pruning the decision tree	2
3.4	Classification rules learning: Concept of classification rules, Separate and conquer approach	2
3.5	The 1R algorithm, Rules from decision trees	1

3.6	Regression methods: Concept of regression, Correlation	1
3.7	Simple linear regression, Ordinary least squares estimation	1
3.8	Multiple linear regression	1
4	**Module 4**	
4.1	Understanding neural networks - Artificial neurons	1
4.2	Activation functions, Network topology	2
4.3	Training neural networks with back propagation	1
4.4	Understanding Support Vector Machines, Classification with Hyperplane	2
4.5	Linearly separable data, Nonlinearly separable data	1
4.6	Methods to find maximum margin hyperplanes in linearly separable data	1
4.7	Using kernels for non-linear spaces	2
5	**Module 5**	
5.1	Understanding Clustering - The k-means clustering algorithm	1
5.2	Using distance to assign and update clusters, Choosing the appropriate number of clusters	1
5.3	Evaluating model performance: Confusion matrices, Precision and recall, Sensitivity and specificity, Precision and recall, F-measure, ROC curves.	2
5.4	Cross validation: K-fold cross validation, Bootstrap sampling	2
5.5	Improving model performance: Bagging, Boosting	1
5.6	Random forests	1

24SJINMCA503	MOBILE COMPUTING	CATEGORY	L	T	P	CREDIT
		CORE	3	1	0	4

Preamble:	The course introduces mobile computing and Android programming fundamentals, enabling students to understand the app development lifecycle and confidently create Android applications independently.
Prerequisite:	24SJINMCA305 Introduction to RDBMS and SQL 24SJINMCA403 Java Programming 24SJINMCA404 Advanced Computer Networks

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the fundamentals of mobile communication technologies and wireless networking standards.	**K3**
CO2	Explain the concepts of mobile computing, its architecture, applications, and various mobile communication systems such as GSM, GPRS, and UMTS.	**K3**
CO3	Compare different mobile operating systems and describe their features, constraints, and responsibilities in mobile environments.	**K3**
CO4	Configure the Android development environment and design effective user interfaces using layouts and views.	**K3**
CO5	Develop basic Android applications using activities, intents, services, and SQLite databases.	**K3**

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	1	2	-	-	-	1
CO2	3	3	2	2	-	-	-	1
CO3	2	3	2	2	-	-	-	1
CO4	2	2	3	3	1	-	-	1
CO5	2	2	3	3	2	1	-	2

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 Marks
Continuous Assessment Test	: 20 Marks
Assignment/Quiz/Course	: 12 Marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1)

1. Explain the components of a wireless communication system.
2. Describe the architecture of mobile telecommunication systems with a neat diagram.
3. Explain the protocol stack of Bluetooth with a diagram.

Course Outcome 2 (CO2)

1. Differentiate between mobile computing and wireless networking.
2. Explain the main features and applications of mobile computing.
3. Compare GSM and UMTS networks with respect to architecture and services.

Course Outcome 3 (CO3)

1. Explain the important features that a mobile operating system must support.
2. Compare the characteristics of Android, Symbian, and Windows Phone operating systems.
3. Illustrate the Android software stack structure with a neat diagram.

Course Outcome 4 (CO4)

1. Describe the Android development environment and its key components.
2. Explain the difference between JVM and DVM.
3. Demonstrate the use of any three layouts in Android with suitable examples.

Course Outcome 5 (CO5)

1. Explain the difference between implicit and explicit intents with examples.
2. Illustrate the activity life cycle of Android with a neat diagram.
3. Describe the procedure to store and retrieve data using SQLite database.

SYLLABUS

Module 1

Basics of Communication Technologies:Mobile handsets, wireless communications and server applications, Components of a Wireless Communication System, Architecture of a Mobile Telecommunication System, Wireless Networking Standards, Wireless Local Area Networks (WLANs), Bluetooth Technology.

Module 2

Introduction to Mobile Computing and Wireless Networking:Mobile Computing vs. Wireless Networking, Mobile Computing Applications, Characteristics of Mobile Computing, Structure of Mobile Computing Application, Cellular Mobile Communication, Global System for Mobile Communication (GSM) – Services, Architecture and Security, General Packet Radio Service (GPRS) – Services, Architecture, Universal Mobile Telecommunications System (UMTS).

Module 3

Operating Systems for Mobile Computing and Mobile Application Protocols: Operating Systems for Mobile Computing – OS Responsibilities in Mobile Devices, Basic Concepts, Special Constraints and Requirements of Mobile OS, Survey of Commercial Mobile OS, Comparative Study of Mobile OS.
Mobile Applications Development and Protocols – Mobile Devices as Web Clients, HDML, WAP, J2ME, Android Application Development.

Module 4

Configuration of Android Environment and User Interface Design: Configuration of Android Environment – Android SDK, Android Development Tools, Android Virtual Devices, Emulators, Dalvik Virtual Machine, Difference between JVM and DVM.
Android User Interface – Linear Layout, Absolute Layout, Frame Layout, Relative Layout, Table Layout. Designing User Interface with Views – TextView, Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton and RadioGroup, ProgressBar, AutoCompleteTextView, Spinner, ListView, ImageView.

Module 5

Activities, Intents, and Database in Android: Activity – Introduction, Intent, Intent-filter, Activity Life Cycle, Broadcast Life Cycle, Service.
SQLite Database in Android – SQLite Database, Creation and Connection of the Database, Extracting Values from a Cursor.

Text Books

1. Prasant Kumar Pattnaik, Rajib Mall, *Fundamentals of Mobile Computing*, Second Edition, PHI (2017).

2. Prasanna Kumar Dixit, *Android*, First Edition, Vikas Publishing (2014).

Reference Books

1. Ashoke K. Talukder, Hasan Ahmed, Roopa R. Yavagal, *Mobile Computing: Technology, Applications and Service Creation*, McGraw Hill.
2. Joseph Annuzzi Jr., Lauren Darcey, Shane Conder, *Introduction to Android Application Development – Android Essentials*, Fifth Edition (2016).
3. Dawn Griffiths & David Griffiths, *Head First Android Development*, First Edition, Shroff Publishers (2015).

MOOC

1. https://nptel.ac.in/courses/106106147
2. https://onlinecourses.swayam2.ac.in/ntr25_ed126/preview

Web Resources

1. https://developer.android.com/index.html
2. https://www.techtarget.com/searchmobilecomputing/definition/nomadic-computing
3. https://www.mpirical.com/technology/4g-lte-lte-advanced-training

Course Contents and Lecture Schedule

Module	Topic	No. of Lectures
Module 1 (09 hrs)	Basics of Communication Technologies – Mobile handsets, wireless communications and server applications	1
	Components of a Wireless Communication System	1
	Architecture of a Mobile Telecommunication System	2
	Wireless Networking Standards	1
	Wireless Local Area Networks (WLANs)	2
	Bluetooth Technology	2
Module 2 (09 hrs)	Introduction to Mobile Computing and Wireless Networking – Mobile Computing vs. Wireless Networking, Mobile Computing Applications, Characteristics of Mobile Computing	2
	Structure of Mobile Computing Application, Cellular Mobile Communication	2
	Global System for Mobile Communication (GSM) – Services, Architecture and Security	2

	General Packet Radio Service (GPRS) – Services, Architecture	2
	Universal Mobile Telecommunications System	1
Module 3 (09 hrs)	Operating Systems for Mobile Computing – OS Responsibilities in mobile devices, A Few Basic Concepts	2
	Special Constraints and Requirements of Mobile OS	1
	Survey of Commercial Mobile OS, A Comparative Study of Mobile OS	2
	Mobile Devices as Web Clients	1
	HDML, WAP	1
	J2ME	1
	Android Application Development	1
Module 4 (10 hrs)	Configuration of Android Environment – Android SDK	1
	Android Development Tools, Android Virtual Devices, Emulators	2
	Dalvik Virtual Machine, Difference between JVM and DVM	1
	Android User Interface: Linear Layout, Absolute Layout, Frame Layout, Relative Layout, Table Layout	3
	Designing UI with Views: TextView, Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton and RadioGroup, ProgressBar, AutoCompleteTextView, Spinner, ListView, ImageView	3
Module 5 (11 hrs)	Activity: Introduction	1
	Intent, Intent-filter	1
	Activity Life Cycle	1
	Broadcast Life Cycle	1
	Service	1
	SQLite Database in Android – SQLite Database	2
	Creation and Connection of the Database	2
	Extracting Values from a Cursor	2

24SJINMCA563	**CYBER SECURITY & CRYPTOGRAPHY**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		ELECTIVE	**3**	**1**	**0**	**4**

Preamble:	This course is designed to provide theoretical concepts used in cryptography and to introduce the students to various cryptographic algorithms and techniques used for implementing data security and protection. This course also discusses common web application security vulnerabilities.
Prerequisite:	Student is expected to have studied Computer Network courses and mathematics courses that cover Elementary Number Theory.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Explain various types of security attacks, security mechanisms, security services and classical encryption techniques.	K2
CO2	Make use of Symmetric and Asymmetric encryption techniques to solve cryptographic problems.	K3
CO3	Describe the concepts of message authentication codes, hash functions and digital signing techniques for ensuring secure transactions.	K2
CO4	Discuss security services in Application, Transport and Network layers.	K2
CO5	Explain common web application security vulnerabilities and various prevention mechanisms.	K2

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	2	1	1				1	1
CO2	2	2	2	1			1	1
CO3	2	1	1				1	1
CO4	2	1	1				1	1
CO5	2	2	1	1			1	1

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 Marks
Continuous Assessment Test (2 numbers)	: 20 Marks
Assignment/Quiz/Course project	: 12 Marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

- Briefly explain each component of OSI security architecture.
- Compare Substitution and Transposition techniques in cryptography.
- Explain how steganography is used in cryptography.

Course Outcome 2 (CO2)

- Explain block cipher modes of operation.
- Compare DES and AES.
- Perform encryption and decryption using RSA Algorithm with parameters: P=17, q = 11, e = 7, M = 88.

Course Outcome 3(CO3):

- Compare the features of HMAC and CMAC algorithms.
- Explain important steps in DSS.
- Describe the terms (a) birthday attack (b) hash cash (c) blind signature

Course Outcome 4 (CO4):

- Explain any one protocol used in E-mail for security.
- Explain how security is provided in Network Layer using IPsec.
- Describe the process of securing electronic transactions.

Course Outcome 5 (CO5):

- Discuss any four Application Security Risks.
- Which are the different forms of XSS and how to prevent these?
- Explain the attack scenario of any four web application security vulnerabilities.

SYLLABUS

Module	Contents	Hours
I	Introduction to Cryptography, OSI security architecture: Security Services, Mechanisms and attacks- Phishing, Ransomware, DoS attack. Network security model. Classical Encryption techniques - Symmetric cipher model, substitution techniques, transposition techniques. Steganography.	7
II	**Conventional Symmetric Key Encryption**: Block ciphers and Stream Ciphers, Block Cipher Design Principles, Modes of operation, Data Encryption Standard, Advanced Encryption Standard (AES), Multiple Encryption, Triple DES. **Public key cryptography:** Principles of public key cryptosystems-The RSA algorithm-Key management – Diffie Hellman Key exchange - Elliptic curve arithmetic - Elliptic curve cryptography.	10
III	**Hash Functions and MAC:** Properties of hash functions, birthday attack, hashcash, Message Authentication Code Algorithms, MAC protocols: HMAC, CMAC. **Digital Signatures**: Classification of signature schemes: RSA signature, Digital Signature Standard, Overview of ElGamal and Schnorr schemes. One time signature schemes, Attacks on Digital Signatures, Blind Signatures.	10
IV	**Introduction to Cyber Security, Email Security**: Security Services for email, Attacks possible through email, Establishing keys privacy, authentication of the source, Message Integrity, Non-repudiation, Pretty Good Privacy, S/MIME. **IP Security:** Overview of IPSec, IPv4 and IPv6, Authentication Header, Encapsulation Security Payload (ESP), Internet Key Exchange. **Transport Level Security**: SSL/TLS Basic Protocol, computing the keys, client authentication, PKI as deployed by SSL, Attacks fixed in v3, Exportability, Encoding, Secure Electronic Transaction (SET).	10
V	**Common web application security vulnerabilities**: Injection flaws, Broken authentication, Sensitive data exposure, XML External Entities (XXE), Broken access control, Security misconfiguration, Cross-Site Scripting (XSS), Insecure deserialization, Using components with known vulnerabilities, Insufficient logging & monitoring.	8

	Example attack scenarios of each of the vulnerabilities listed; how to prevent them.	

Text Books

1. William Stallings, “Cryptography and Network Security,” 6th Edition, Pearson Education.
2. Behrouz A. Forouzan, “Introduction to Cryptography and Network Security”, Tata McGraw-Hill Publishing 2nd Edition.

Reference Books

1. Charlie Kaufman, Radia Perlman and Mike Speciner, “Network Security”, Prentice Hall of India.
2. Manuel Mogollon, “Cryptography and Security Services – Mechanisms and Applications”, Cybertech Publishing.
3. William R. Cheswick, Steven M. Bellovin, Aviel D. Rubin, “Firewalls and Internet Security” Addison- Wesley.

Web References

1. http://www.hashcash.org/hashcash.pdf [Reference for hash cash]
2. https://www.coursera.org/learn/crypto
3. https://www.khanacademy.org/computing/computer-science/cryptography

Course Contents and Lecture Schedule

No.	Topic	No. of Lecture
1	Introduction to Cryptography	**7 Hours**
1.1	What is cryptography, Related Terms, Need of cryptosystems	1
1.2	OSI security architecture: Security Services, Mechanisms	1

1.3	Security attacks- Phishing, Ransomware, DoS attack.	1
1.4	Network security model	1
1.5	Classical Encryption techniques, Symmetric cipher model	1
1.6	Substitution techniques	1
1.7	Transposition techniques, Steganography	1
2	**Conventional Symmetric and Public Key Encryption**	**10 Hours**
2.1	Block ciphers and Stream Ciphers, Block Cipher Design Principles	1
2.2	Modes of operation	1
2.3	Data Encryption Standard	1
2.4	Advanced Encryption Standard (AES)	1
2.5	Multiple Encryption, Triple DES	1
2.6	Public key cryptography: Principles of public key cryptosystems	1
2.7	The RSA algorithm	1
2.8	Key management	1
2.9	Diffie Hellman Key exchange	1
2.10	Elliptic curve arithmetic - Elliptic curve cryptography.	1
3	**Hash Functions and MAC**	**10 Hours**
3.1	Properties of hash functions, birthday attack	1
3.2	Hashcash, Message Authentication Code Algorithms	1
3.3	MAC protocols: HMAC, CMAC	1
3.4	Digital Signatures: Classification of signature schemes	1
3.5	RSA signature	1

3.6	Digital Signature Standard	1
3.7	Overview of ElGamal and Schnorr schemes	1
3.8	One time signature schemes	1
3.9	Attacks on Digital Signatures	1
3.10	Blind Signatures	1
4	**Introduction to Cyber Security**	**10 Hours**
4.1	Email Security: Security Services for email, Attacks possible through email	1
4.2	Establishing keys privacy, authentication of the source, Message Integrity, Non-repudiation	1
4.3	Pretty Good Privacy, S/MIME	1
4.4	IP Security: Overview of IPSec	1
4.5	IPv4 and IPv6, Authentication Header	1
4.6	Encapsulation Security Payload (ESP), Internet Key Exchange	1
4.7	Transport Level Security: SSL/TLS Basic Protocol	1
4.8	computing the keys, client authentication, PKI as deployed by SSL	1
4.9	Attacks fixed in v3, Exportability, Encoding	1
4.10	Secure Electronic Transaction (SET)	1
5	**Common web application security vulnerabilities**	**8 Hours**
5.1	Common web application security vulnerabilities	1
5.2	Injection flaws, Broken authentication	1
5.3	Sensitive data exposure, XML External Entities (XXE)	1
5.4	Broken access control, Security misconfiguration	1

5.5	Cross-Site Scripting (XSS), Insecure deserialization	1
5.6	Using components with known vulnerabilities, Insufficient logging & monitoring.	1
5.7	Example attack scenarios of each of the vulnerabilities listed	1
5.8	How to prevent each of the vulnerabilities.	1

24SJINMCA565	CLOUD COMPUTING	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble:	The syllabus is prepared with a view to equip the students to learn basic concepts in cloud computing - compute, storage, networking. They should gain basic understanding of orchestration, HA and failover.
Prerequisite:	Awareness in Virtualisation and Containers is desirable.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Understand the basic concepts in cloud computing and OpenStack logical architecture	K2
CO2	Discuss OpenStack cloud controller and common services	K3
CO3	Compare different OpenStack compute service components and storage types	K2
CO4	Describe the OpenStack Networking- Connection types and networking services	K2
CO5	Discuss orchestration, HA and failover in OpenStack	K2

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2			2				2
CO2	3	2		2				2
CO3	2	2		2				2
CO4	2	2		2				2
CO5	2	2	2	2				2

Mark Distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks

Continuous Assessment Test (2 numbers)	: 20 Marks
Assignment/Quiz/Course Project	: 12 Marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum 2 subdivisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. List and explain various components of Nova compute service.
2. Explain the neutron architecture?
3. Briefly describe keystone identity management.

Course Outcome 2 (CO 2):

1. Explain the telemetry services in OpenStack.
2. Explain the steps involved in bringing up a working OpenStack Ansible on the deployment host.
3. Explain the steps in network configuration

Course Outcome 3 (CO 3):

1. Explain briefly swift architecture
2. Briefly explain how data is handled in the cluster by swift
3. What is meant by CPU over commitment?

Course Outcome 4 (CO 4):

1. Explain steps in associating a floating IP to a virtual machine.
2. Briefly explain the steps in creating a virtual network with two subnets
3. Briefly explain Linux bridge-based connectivity?

Course Outcome 5 (CO 5):

1. Briefly explain the major components in heat?
2. Explain the different metrics that can be measured in a highly available infrastructure?
3. Explain the need for Service level agreement

SYLLABUS

Module 1: Overview of OpenStack (7 Hours)
Introduction to cloud computing, private cloud, public cloud, hybrid cloud architecture. Cloud Services - Infrastructure as a Service, Platform as a Service, Storage as a Service. Designing OpenStack Cloud Architectural Consideration - OpenStack - The new data centre paradigm - OpenStack logical architecture - Nova - Compute Service-Neutron - Networking services - Gathering the pieces and building a picture - A sample architecture setup.
Module 2: OpenStack cluster - Controller and common services (6 Hours)
OpenStack Cluster – The Cloud Controller and Common Services- Asymmetric clustering, Symmetric clustering, The cloud controller - The keystone service. The nova-conductor service, The nova-scheduler service, The API services, Image management, The network service, The horizon dashboard, The telemetry services.
Module 3: OpenStack compute and Storage (12 Hours)
OpenStack Compute -The compute service components - Deciding on the hypervisor - OpenStack Magnum Project - Segregating the compute cloud - Overcommitment considerations - Storing instances' alternatives - Understanding instance booting - Planning for service recovery. OpenStack Storage - Block, Object, and File Share - Understanding the storage types - Ephemeral Storage - Persistent storage - A spotlight on Swift - Deploying Swift service - Using block storage service: Cinder.
Module 4: OpenStack Networking (10 Hours)
The architecture of Neutron - Implementing virtual networks - Connecting virtual networks with routers - Implementing network security in OpenStack. OpenStack Networking - The architecture of Neutron - Implementing virtual networks - VLAN, Tunnel based, Virtual Switches, The ML2 Plugin. Neutron Subnets - Connecting virtual networks with routers - Configuring the routing service - connecting networks using a virtual router, connecting to the external world, connectivity from the external world, associating a floating IP - Implementing network security in OpenStack
Module 5: OpenStack Orchestration, HA and failover (10 Hours)
Orchestration in OpenStack - Heat and its Components, stacking in OpenStack, OpenStack Orchestration with Terraform. OpenStack HA and failover: Scope of HA in OpenStack, HA in the database, HA in the Queue, Implementing HA on RabbitMQ.

Text Book

1. Omar Khedher, Chandan Datta Chowdhury, Mastering OpenStack, 2nd Edition, Packt Publishing, 2017

Reference Books

1. Tom Fifield, Diane Fleming, Anne Gentle,Lorin Hochstein, Jonathan Proulx, Everett Toews, and Joe Topjian, OpenStack Operations Guide, O'REILY,1/e, 2014.
2. Uchit Vyas, Applied OpenStack Design Patterns, Apress, 1/e, 2016.
3. V. K. Cody Bumgardner, OpenStack in action, Manning, 2016.
4. Amar Kapadia, Sreedhar Varma, Kris Rajana, Implementing Cloud Storage with OpenStack Swift, Packt Publishing, 2014.
5. https://docs.openstack.org/wallaby/?_ga=2.231002015.1428061357.1620834394-1139122985.1620834394

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Overview of OpenStack**	**7 Hours**
1.1	Introduction to cloud computing, private cloud, public cloud, hybrid cloud architecture.	1
1.2	Cloud Services - Infrastructure as a Service, Platform as a Service, Storage as a Service	1
1.3	Designing OpenStack Cloud Architectural Consideration - OpenStack - The new data center paradigm -OpenStack logical architecture	1
1.4	Nova - Compute service	1
1.5	Neutron - Networking services	1
1.6	Gathering the pieces and building a picture	1
1.7	A sample architecture setup	1
2	**OpenStack cluster - Controller and common services**	**6 Hours**
2.1	OpenStack Cluster – The Cloud Controller and Common Services- Asymmetric clustering, Symmetric clustering	1
2.2	The cloud controller - The keystone service	2
2.3	The nova-conductor service, The nova-scheduler service, The API services, Image management.	1

2.4	The network service, The horizon dashboard, The telemetry services	2
3	**OpenStack compute and Storage**	**12 Hours**
3.1	The compute service components-Deciding on the hypervisor-OpenStack Magnum project	1
3.2	Segregating the compute cloud	1
3.3	Overcommitment considerations	1
3.4	Storing instances' alternatives	1
3.5	Understanding instance booting	1
3.6	Planning for service recovery	1
3.7	OpenStack Storage - Block, Object, and File Share-Understanding the storage types	1
3.8	A spotlight on swift	2
3.9	Deploying swift service	1
3.10	Using Block Storage Service Cinder	2
4	**OpenStack Networking**	**10 Hours**
4.1	The architecture of Neutron	1
4.2	Implementing virtual networks - VLAN, Tunnel based	1
4.3	Virtual Switches, The ML2 Plugin	1
4.4	Neutron Subnets	2
4.5	Connecting virtual networks with routers - Configuring the routing service	1
4.6	Connecting networks using a virtual router, Connecting to the externa world	1
4.7	Connectivity from the external world, Associating a floating IP	1
4.8	Implementing network security in OpenStack	2
	OpenStack Orchestration, HA and Failover	**10 Hours**
5.1	Orchestration in OpenStack, Heat and its Components	1
5.2	Stacking in OpenStack	2
5.3	OpenStack Orchestration with Terraform	2

5.4	Scope of HA in OpenStack	2
5.5	HA in the database	1
5.6	HA in the Queue, Implementing HA on RabbitMQ	2

Suggested Assignments

1) Create VMs in your physical machine using OpenStack to set up the following services: Moodle, MySQL Server, Samba. Design the desired configuration of the physical machine to handle the requirements of the entire college.
2) Set up storage services for storing external files for Moodle.
3) Set up firewall rules for samba, MySQL server, allow the connection to MySQL server only to Moodle VM.
4) Set up recovery plans for the above services
5) Convert the MySQL server to HA MySQL server.
6) Setup a load balancer for the Moodle server.

		CATEGORY	L	T	P	CREDIT
24SJINMCA581	**INTERNET OF THINGS**	**ELECTIVE**	3	1	0	4

Preamble:	This course intends to provide insight into new innovations that will build novel type of interactions among things and humans, and enables the realization of smart cities, infrastructures, and services for enhancing the quality of life and utilization of resources. An overview of IoT and its related concepts, different IoT architectures and their components, emerging paradigms such as Fog computing, Platforms and solutions supporting development and deployment of IoT applications, message passing mechanisms such as RPC, REST, and CoAP, data and knowledge management, data confidentiality, data integrity, and operation control issues faced by IoT are included in the course.
Prerequisite:	Basic concepts of Information Technology and Internet.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Describe the main concepts and features of the IoT paradigm.	K2
CO2	Discuss Fog computing, TinyOS - nesC and programming frameworks for IoT	K2
CO3	Describe the data management techniques applied to the IoT environment.	K3
CO4	Explain security, and privacy in IOT environments	K2
CO5	Discuss key enablers and solutions to enable practical IoT systems	K3

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	0	2	2				2
CO2	3	0	2	3				2
CO3	3	0	3	3	1			2
CO4	2	0	3	3	1			2
CO5	3	1	3	3	2	2		3

Mark Distribution

Total Marks	CIA	ESE	ESE Duration
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks
Continuous Assessment Test (2 numbers) : 20 Marks
Assignment/Quiz/Course project : 12 Marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 subdivisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1)

1. Compare SOA-based architecture and API-oriented architecture.
2. Neatly sketch the open IoT architecture for IoT/CLOUD convergence.
3. List and explain the applications of device/cloud collaboration.

Course Outcome 2 (CO2)

1. What are the advantages associated with Fog computing?
2. Comment on the four broad requirements that motivate the design of TinyOS.
3. Summarize the communication paradigms and technologies used in resource-constrained environments.

Course Outcome 3(CO3)

1. Explain stream and stream processing in IoT.
2. Write and explain the algorithm for distributed anomaly detection by clustering ellipsoids.
3. Discuss the general architecture of a stream-processing system in IoT.

Course Outcome 4 (CO4)

1. Give an overview on the security requirements of IoT.
2. How can you nullify the impact of fault in high-availability cluster?
3. Explain the BCK with pre-shared keys for TinyTO.

Course Outcome 5 (CO5)

1. Give an overview on the Wired Gateway Interfaces.

2. List the features to select the gateway hardware.
3. List the steps to prepare Raspberry Pi for the execution.

SYLLABUS

Module 1 (9 Hours)

Overview of Internet of Things: Open-source semantic web infrastructure for managing IoT resources in the Cloud - Device/Cloud Collaboration framework for intelligence applications.

Module 2 (11 Hours)

Introduction to Fog Computing: principles, architectures, and applications. TinyOS - NesC, Programming frameworks for Internet of Things

Module 3 (8 Hours)

Stream processing in IoT: foundations, state-of-the-art, and future directions - A framework for distributed data analysis for IoT

Module 4 (9 Hours)

Security and privacy in the Internet of Things- Internet of Things - robustness and reliability. TinyTO: two-way authentication for constrained devices in the Internet of Things - Obfuscation and diversification for securing the Internet of Things

Module 5 (8 Hours)

Creating a simple IoT project - Preparing Raspberry Pi Interfacing the hardware – Internal representation of sensor values- Persisting data - Creating the actuator project - Creating a controller.

More detailed knowledge may be acquired through seminars, assignments and talks by eminent external experts and also by implementing a micro project.

Any one of the following or similar micro projects may be given as part of the course.

1. Smart Gas Leakage Detector

2. Night Patrol at home

Text Books
1. RajkumarBuyya; Amir VahidDastjerdi ,

Reference Books
S. SitharamaIyengar; Nandan Parameswaran; Vir V. Phoha; N. Balakrishnan; Chuka Okoye, December 14, 2010

Web Resources
1. https://www.coursera.org/specializations/internet-of-things
2. http://web.mit.edu/professional/digital-programs/courses/IoT

Course Contents and Lecture Schedule

No	Topic	No. of Lectures (Hours)
1	**Introduction**	**9**
1.1	Overview of Internet of Things	3
1.2	Open-source semantic web infrastructure for managing IoT resources in the Cloud	3
1.3	Device/Cloud Collaboration framework for intelligence applications.	3
2	**Programming frameworks**	**11**
2.1	Introduction to Fog Computing	2
2.2	principles, architectures, and applications	3
2.3	TinyOS - NesC	3
2.4	Programming frameworks for Internet of Things	3
3	**Data management techniques**	**8**
3.1	Stream processing in IoT	2
3.2	foundations, state-of-the-art, and future directions	3
3.3	A framework for distributed data analysis for IoT	3
4	**Security and privacy**	**9**
4.1	Security and privacy in the Internet of Things	2

4.2	Internet of Things - robustness and reliability	2
4.3	TinyTO: two-way authentication for constrained devices in the Internet of Things	2
4.4	Obfuscation and diversification for securing the Internet of Things	3
5	**IoT Implementation**	**8**
5.1	Creating a simple sensor project	2
5.2	Preparing Raspberry Pi	1
5.3	Interfacing the hardware ----	1
5.4	Internal representation of sensor values	1
5.5	Persisting data	1
5.6	Creating the actuator project	1
5.7	Creating a controller	1

<table>
<tr><td rowspan="2">24SJINMCA583</td><td rowspan="2">DEEP LEARNING</td><td>CATEGORY</td><td>L</td><td>T</td><td>P</td><td>CREDIT</td></tr>
<tr><td>ELECTIVE</td><td>3</td><td>1</td><td>-</td><td>4</td></tr>
</table>

Preamble:	This course intends to provide insight into deep learning. This topic is currently a much sought-after skill and is under active research. Students have to refer appropriate research papers and multiple books to get in-depth knowledge about the topics. Instructors may give suitable programming assignments to augment the material covered in the classroom
Prerequisite:	Basic concepts of linear algebra, probability and optimization.

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Explain the basic concepts of deep learning.	K2
CO2	Design neural networks using TensorFlow	K3
CO3	Solve real world problems with CNN.	K3
CO4	Solve real world problems with RNN.	K3
CO5	Describe the concepts of GAN.	K2

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	2	1	1	1	2
CO2	3	2	3	3	2	1	1	2
CO3	3	3	3	3	2	2	1	3
CO4	3	3	3	3	2	2	1	3
CO5	2	2	2	3	1	1	1	2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 Hours

Continuous Internal Evaluation Pattern:

Attendance : 8 Marks

Continuous Assessment Test (2 numbers) : 20 Marks

Assignment/Quiz/Course project : 12 Marks

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Course Level Assessment

Course Outcome C01:

1. Describe the model of a biological neuron.
2. Explain Perceptron learning algorithm.
3. Explain the role of batch normalization in training a neural network

Course Outcome CO2 :

1. Draw and demonstrate the VGG-16 architecture.
2. Sketch the AlexNet architecture and explain its functionalities.

Course Outcome CO3 :

1. Design a convolutional neural network which can classify MNIST handwritten data.
2. An input image has been converted into a matrix of size 12 X 12 along with a filter of size 3 X 3 with a Stride of 1. Determine the size of the convoluted matrix.
3. Why do we prefer Convolutional Neural networks (CNN) over Artificial Neural networks (ANN) for image data as input?

Course Outcome CO4 :

1. You are given an image data set with 10 classes. Describe how you will use deep learning to build a classifier.
2. Design a system to generate deep fakes from an image.

Course Outcome CO5 :

1. Describe auto encoders and how they help in dimensionality reduction.
2. Explain how GANS work.

SYLLABUS

Module I

Review of Neural Networks: Model of a biological neuron, McCulloch Pitts Neuron, Activation Functions, Perceptron, Perceptron Learning Algorithm and Convergence, Multilayer Perceptron, Back propagation, Learning XOR, Sigmoid Neurons, Gradient Descent, Feed forward Neural Networks.

Module II

Training Neural Networks: Initialization, dropout, batch normalization and dropout, overfitting, underfitting, training and validation curves.

Data Visualization: Feature and weight visualization, tSNE.

Introduction to TensorFlow: graphs, nodes, Tensor data structures - rank, shape, type, Building neural networks with TensorFlow, Introduction to Keras.

Module III

Convolutional Neural Networks: Convolution operation, Convolutional layers in neural network, pooling, fully connected layers.

Case study: Architecture of Lenet, Alexnet and VGG 16

Module IV

Recurrent Neural Networks: Back propagation, vanishing gradients, exploding gradients, truncated backpropagation through time, Gated Recurrent Units (GRUs), Long Short- Term Memory (LSTM) cells, solving the vanishing gradient problem with LSTMs.

Module V

Autoencoders, variational autoencoders.

Generative Adversarial Networks (GAN): Discriminative and generative models, GAN discriminator, GAN generator, upsampling, GAN Training, GAN challenges, loss functions, cross entropy, minimax loss, Wasserstein loss.

Text Books

1. Generative Deep Learning: David Foster, OReily, (2019)
2. Deep Learning, Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT press(2016)
3. Hands on Machine Learning with Scikit Learn and TensorFlow, Aurélien Géron (2019)

4. Deep Learning Illustrated, Jon Krohn, Grant Beyleveld, Aglae Bassens, Pearson, 1st Edn., (2020)
5. Online book Dive Deep into Machine Learning at https://d2l.ai

References

Module 1

a. https://www.cse.iitm.ac.in/~miteshk/CS6910/Slides/Lecture2.pdf
b. https://www.cse.iitm.ac.in/~miteshk/CS6910/Slides/Lecture3.pdf

Module 2

a. http://neuralnetworksanddeeplearning.com
b. Hands on Machine Learning with Scikit Learn and TensorFlow, Aurélien Géron
c. Probabilistic Machine Learning: An Introduction, Kevin Murphy
d. https://www.researchgate.net/publication/228339739_Vualizing_data_using_t- SNE

Module 3

a. https://www.cse.iitm.ac.in/~miteshk/CS7015/Slides/Teaching/pdf/Lecture11.pdf
b. Convolutional neural networks for visual computing (Chapter 4), Ragav Venkatesan and Baoxin Li CRC press

Module 4

a. On the difficulty of training RNNs: https://arxiv.org/pdf/1211.5063.pdf
b. LSTM: A Search Space Odyssey: https://arxiv.org/abs/1503.04069
c. Understanding Deriving and Extending the LSTM: https://r2rt.com/written-memories-understanding-deriving-and-extending-the-lstm.html
d. Understanding-LSTMs/
https://www.cse.iitm.ac.in/~miteshk/CS7015/Slides/Teaching/pdf/Lecture14.pdf
https://www.cse.iitm.ac.in/~miteshk/CS7015/Slides/Teaching/pdf/Lecture15.pdf

Module 5

a. GANs in Action: Deep Learning with Generative Adversarial Network
Jakub Langgr, Vladimir Bok
b. Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play
David Foster
c. https://developers.google.com/machine-learning/gan

Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**8 Hours**
1.1	Review of Neural Networks: Model of a biological neuron	1
1.2	McCulloch Pitts Neuron, Activation functions	1
1.3	Perceptron, Perceptron Learning Algorithm	1
1.4	Convergence, Multilayer Perceptron	1
1.5	Back propagation	1
1.6	Learning XOR, Sigmoid Neurons	1
1.7	Gradient Descent, Feed forward Neural Networks	2
2	**Module 2**	**10 Hours**
2.1	Training Neural Networks	1
2.2	Initialization, Dropout	1
2.3	Batch normalization and drop out	1
2.4	Over fitting, under fitting, training and validation curves, data visualization, feature and weight visualization, tSNE	2
2.5	Introduction to TensorFlow, graphs, nodes, Tensor Data Structures - rank, shape, type	2
2.6	Building neural networks with tensor flow	2
2.7	Introduction to Keras	1
3	**Module 3**	**10 Hours**
3.1	Convolutional neural networks	1
3.2	Convolution operation	2
3.3	Back propagation in multilayer neural networks	1
3.4	Convolutional layers in neural network, pooling	2
3.5	Fully connected layers	2
3.6	Case study: Architecture of Lenet, Alexnet and VGG 16	2
4	**Module 4**	**8 Hours**
4.1	Recurrent neural networks	1

4.2	Back propagation: vanishing gradients, exploding gradients	1
4.3	Truncated Backpropagation Through Time	1
4.4	LSTM	1
4.5	Gated Recurrent Units (GRUs)	1
4.6	Long Short-Term Memory (LSTM) Cells	1
4.7	Solving the vanishing gradient problem with LSTMs	2
5	**Module 5**	**9 Hours**
5.1	Autoencoders, Variational autoencoders	2
5.2	Generative Adversarial Networks (GAN)	1
5.3	Discriminative and generative models	2
5.4	GAN Discriminator, GAN Generator, upsampling,	1
5.5	GAN Training	1
5.6	GAN challenges, Loss functions, cross entropy, minimax loss, Wasserstein loss	2

24SJINMCA509	MINI PROJECT 2	CATEGORY	L	T	P	CREDIT
		PROJECT	0	0	4	2

Preamble:	The mini project is designed to develop practical ability and knowledge about tools/techniques in order to solve the actual problems related to the industry, academic institutions or similar areas.
Prerequisite:	Knowledge in software engineering principles and programming skills

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Practice acquired knowledge within the chosen area of technology for project	K3
CO2	Identify, discuss and justify the technical aspects of the chosen project with a comprehensive and systematic approach.	K4
CO3	Reproduce, improve and refine technical aspects for engineering projects.	K4
CO4	Work as an individual or in a team in development of technical projects.	K3
CO5	Communicate and report effectively project related activities and findings	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	2	1	1	1	2
CO2	2	3	3	2	1	2	1	2
CO3	2	2	3	3	1	2	1	2
CO4	1	1	2	1	3	2	1	2
CO5	1	2	2	1	3	2	1	2

SYLLABUS

Students can take up any application level/system level experimental design/implementation tasks of relatively minor intensity and scope as compared to the major-project, pertaining to a relevant domain of study. Projects can be chosen either from the list provided by the faculty or in the field of interest of the student. At the end of each phase, presentation and demonstration of the project should be conducted, which will be evaluated by a panel of examiners. A detailed project report duly approved by the guide and Head of the department in the prescribed format should be submitted by the student for final evaluation. Students as individual or a team of maximum 4 members may be trusted with providing solution to similar problems

Mini project will give an opportunity for the students to prepare for the main project and to achieve some of the objectives of the main project. This can also be used as an opportunity for producing and distributing socially useful software's.

Internal Continuous Assessment (100 marks)

10% (10) - Project Synopsis / Proposal Evaluation

20% (20) - 1st Interim Project Evaluation

20% (20) - 2nd Interim Project Evaluation

50% (50) - End Semester Internal Project work & Project Report Evaluation by the Panel Constituted by the Head of the department.

General Guidelines:

● Students shall identify Real-Life Projects which are relevant and useful to the society or industry.
● The project shall be an individual/ Group (Maximum of 4 members) project and must be done in-house. The student has to spend time in the lab for the project work. Attendance as per MCA regulations is applicable for submitting the project for final evaluation.
● However, in exceptional cases, students may be permitted to undertake the project work at industry premises, if offered by the organization. The company proposal should be approved by the Department Internship Scrutiny Committee, constituted by the Head of the Department. This internship shall be completed during the semester break period between Semester 5 and Semester 6.

- In such cases, the student is required to produce a letter from the organization before starting the project and an internship scrutiny committee constituted by the head of the department shall make the decision on permission. Industries and training institutes that offer project work for a fee shall not be permitted.

● Students shall submit project synopsis and get prior approval from the Project (Faculty) Supervisor before the project work begins.
● If there is a customer for the project then he/she will be the Product Owner (External Guide) and a faculty from the department will be the Internal Guide. If there is no such customer then the Internal Guide himself/herself shall act as the Product Owner.
● A faculty / technical staff shall act as the Scrum Master to continuously monitor the project development. Periodic meetings, of less than 15 minutes, at the convenience of the Scrum Master are to be highly encouraged. Ensure such meetings occur once in three days.
● The student shall maintain a Scrum Book (Rough Record) which has to be divided into 4 parts – (i) Product Backlog (ii) Database & UI Design (iii) Testing & Validation and (iv) Details of Versions. Make dated entries in the corresponding part at regular intervals. The corrections and comments from Product Owner and Scrum Master should be clearly indicated with the Date.
● Test Driven Development methodology may be practiced for the project development. BugZilla, BackLog or any such tool may be used for Bug Tracking.
● Git shall be used for Version Control and Git commit history may be verified as part of project evaluation.
● LaTeX or an equivalent tool shall be used for preparing Presentations and Project Report.

● Students shall be encouraged to publish their work in journals and due credit shall be given to such students.
● For the externally done projects, periodic confidential progress report and attendance statement shall be collected from the External Guide and be reviewed by the Project Supervisor.
● Set a sprint as two weeks, ensure biweekly reviews. A review shall not exceed 30 minutes. A demo to the Product Owner (Project Guide) is mandatory in every review.
● Interim evaluations of the project's progress shall be conducted by a Project Assessment Board as part of internal assessment. Two such evaluations are desirable. Scrum reviews shall not be sacrificed for such presentations.
● The Project Assessment Board shall be constituted by the Head of the Department with the following three members.

Chairman:
1. Head of the Department
Members:
2. Project supervisor/s of the student
3. One Senior faculty member from the Department

● At the end of the semester, Project Assessment Board evaluation shall be there on the entire project development activities.

Week	Schedule
(May be scheduled in line with the academic calendar)	
1	Selection of Topic, Submission of project synopsis and getting approval Meeting of Development Team including Scrum Master with Product Owner (Project Guide)
2	Commencement of the Project.
3	First Sprint release and Scrum Review by the Product Owner (Project Guide)
4	Second Sprint release and Scrum Review by the Project Guide First interim evaluation by the Project Assessment Board
5	Third Sprint release and Scrum Review by the Project Guide
6	Fourth Sprint release and Scrum Review by the Project Guide
7	Second interim evaluation by the Project Assessment Board
8	Fifth Sprint release and Scrum Review by the Project Guide
9	Submission of project report, with Scrum Book Final project presentation Evaluation by the Project Assessment Board

24SJINMCA531	DATA SCIENCE LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble:	This is an introductory practical course on Data Science and student will learn how to use various scientific libraries in python to implement data mining techniques and machine learning algorithms.
Prerequisite:	Fundamentals of programming, python programming fundamentals, Machine learning, fundamentals of web programming

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Use different python packages to perform numerical calculations, statistical computations and data visualization	Apply (K3)
CO2	Use different packages and frameworks to implement regression and classification algorithms.	Apply (K3)
CO3	Use different packages and frameworks to implement text classification using SVM and clustering using k-means	Apply (K3)
CO4	Use popular deep learning frameworks to develop and train neural network models and evaluate the performance of different models	Apply (K3)
CO5	Implement programs for web data mining and natural language processing using NLTK	Apply (K3)

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	3	-	-	-	3
CO2	3	3	2	3	-	-	-	3
CO3	3	3	2	3	-	-	-	3
CO4	3	3	2	3	-	-	-	3
CO5	3	3	2	3	-	-	-	3

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 Hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	7.5
Maintenance of daily lab record and GitHub management	10
Regular class viva voce	7.5
Timely completion of day-to-day tasks	10
Tests/Evaluation	15

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 Marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 Marks</td></tr>
<tr><td rowspan="4">Problem solving
(Based on difficulty level, one or more questions may be given)</td><td>Flowchart / Algorithm / Structured
description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 Marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output</td><td>20%</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1)

- Review of python programming – Programs review the fundamentals of python (simple python programs ice breaker) – (at most one lab session)

- Matrix operations (using vectorization) and transformation using python and SVD using Python.
- Programs using matplotlib / plotly / bokeh / seaborn for data visualisation.
- Programs to handle data using pandas

Course Outcome 2 (CO2)

- Program to implement k-NN classification using any standard dataset available in the public domain and find the accuracy of the algorithm.
- Program to implement Naïve Bayes Algorithm using any standard dataset available in the public domain and find the accuracy of the algorithm.
- Program to implement linear and multiple regression techniques using any standard dataset available in the public domain and evaluate its performance.

Course Outcome 3 (CO3):

- Program to implement text classification using Support vector machine.
- Program to implement decision trees using any standard dataset available in the public domain and find the accuracy of the algorithm.
- Program to implement k-means clustering technique using any standard dataset available in the public domain

Course Outcome 4 (CO4):

- Program to implement multilevel perceptron for classification (Iris Dataset).
- Program to implement Feedforward Neural Network for Binary Classification using activation functions like Sigmoid and ReLU.

Course Outcome 5 (CO5)

<u>Web Data Mining:</u>

- Implement a simple web crawler (ensure ethical conduct).
- Implement a program to scrap the web page of any popular website – suggested python package is scrappy (ensure ethical conduct).

<u>Natural Language Processing:</u>

Problems may be designed for the following topics so that students can get hands on experience in using python for natural language processing:

- Part of Speech tagging
- N-gram and smoothening
- Chunking

SYLLABUS
Review of python programming, Matrix operations, Data Visualisation using matplotlib /plotly / bokeh / seaborn, Data handling using pandas, Classification k-NN algorithm, Naïve Bayes algorithm, Implementation of linear and multiple regression techniques, Text classification using Support vector machine, Implementation of Decision Trees, Clustering using k-means algorithm, Convolutional Neural Network to classify images using Keras framework, Web Crawler and Scrapping web pages, Implementation of NLP - Part of Speech tagging, N-gram & smoothening and Chunking using NLTK.

Reference Books

1. Christopher M Bishop, “Pattern Learning and Machine Learning”, Springer, 2006
2. E. Alpayidin, “Introduction to Machine Learning”, Prentice Hall of India (2005)
3. T. Hastie, RT Ibrashiran and J. Friedman, “The Elements of Statistical Learning”, Springer 2001
4. Toby Segaran, “Programming Collective Intelligence: Building Smart Web 2.0 Applications”, O' Reilly Media; 1 edition (16 August 2007).
5. Drew Conway, John Myles White, “Machine Learning for Hackers: Case Studies and
6. Algorithms to Get You Started”, O' Reilly Media; 1 edition (13 February 2012)
7. Simon Rogers, Mark Girolami, “A First course in Machine Learning”, CRC Press, First Indian reprint, 2015.
8. Tom Mitchell, “Machine Learning”, McGraw Hill, 1997.
9. Bing Liu, Web Data Mining - Exploring Hyperlinks, Contents and Usage Data_ Second edition, Springer 2011

Course Contents and Lab Schedule:

Sl. No	Topic	No: of Hours
1	Review of python programming, Matrix operations, Programs using matplotlib / plotly / bokeh / seaborn for data visualisation and programs to handle data using pandas.	8
2	Program to implement k-NN classification using any standard dataset available in the public domain and find the accuracy of the algorithm	2
3	Program to implement Naïve Bayes Algorithm using any standard dataset available in the public domain and find the accuracy of the algorithm	4
4	Program to implement linear and multiple regression techniques using any standard dataset available in the public domain and evaluate its performance.	4
5	Program to implement text classification using Support vector machine.	4
6	Program to implement decision trees using any standard dataset available in the public domain and find the accuracy of the algorithm	4
7	Program to implement k-means clustering technique using any standard dataset available in the public domain	4
8	Program on multilayer and feed forward neural network to classify data using any data set available in public domain	6
9	Program to implement a simple web crawler and scrapping web pages.	6
10	Implement problems on natural language processing - Part of Speech tagging, N-gram & smoothening and Chunking using NLTK	4

24SJINMCA533	MOBILE APPLICATION DEVELOPMENT LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble:	This is a practical course on Mobile Application Development and student will learn how to program in Android Platform and develop applications using SQLite that run on Android Operating System.
Prerequisite:	Basic knowledge on programming and database concepts.

Course Outcomes: After the completion of the course the student will be able to		**K Level**
CO1	Design and develop user interfaces for mobile apps using basic building blocks, UI components and application structure using Emulator	K3
CO2	Write simple programs and develop small applications using the concepts of UI design, layouts and preferences	K3
CO3	Develop applications with multiple activities using intents, array adapter, exceptions and options menu.	K3
CO4	Implement activities with dialogs, spinner, fragments and navigation drawer by applying themes	K3
CO5	Develop mobile applications using SQLite.	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	2			2
CO2	3	3	3	3	2			2
CO3	3	3	3	3	2			2
CO4	3	3	3	3	2			2
CO5	3	3	3	3	2	2		2

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 Hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	7½
Maintenance of daily lab record and GitHub management	10
Regular class viva voce	7½
Timely completion of day-to-day tasks	10
Tests/Evaluation	15

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 Marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 Marks</td></tr>
<tr><td rowspan="4">Problem solving (Based on difficulty level, one or more questions may be given)</td><td>Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 Marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output</td><td>20%</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Design a Login Form with username and password using LinearLayout and toast valid credentials
2. Write a program that demonstrates Activity Lifecycle.
3. Implementing basic arithmetic operations of a simple calculator
4. Implement validations on various UI controls

Course Outcome 2 (CO2)

1. Design a registration activity and store registration details in local memory of phone using Intents and SharedPreferences
2. Design a simple Calculator using GridLayout and Cascaded LinearLayout
3. Create a Facebook page using RelativeLayout; set properties using .xml file
4. Develop an application that toggles image using FrameLayout

Course Outcome 3(CO3):

1. Implement Adapters and perform exception handling
2. Implement Intent to navigate between multiple activities
3. Develop application that works with explicit intents
4. Implement Options Menu to navigate to activities
5. Develop an application that uses ArrayAdapter with ListView.

Course Outcome 4 (CO4):

1. Develop an application that use GridView with images and display Alert box on selection
2. Develop an application that implements Spinner component and perform event handling
3. Apply themes via code and manifest file
4. Develop application using Fragments
5. Implement Navigation drawer

Course Outcome 5 (CO5):

1. Create database using SQLite and perform INSERT and SELECT
2. Perform UPDATE and DELETE on SQLite database
3. Develop an application as a micro project which uses SQLite database as an assignment

SYLLABUS

Fundamentals: Basic Building blocks – Activities, Services, Broadcast Receivers and Content providers, UI Components – Views and notifications Components for communication -Intents and Intent Filters
Application Structure: AndroidManifest.xml, user-permission – sdk, Resources and R.java, Assets, Layouts and Drawable Resources, Activities and Activity lifecycle.
Emulator-Android Virtual Device: Launching emulator, Editing emulator settings, Emulator shortcuts, Logcat usage, Introduction to DDMS
Basic UI design: Form widgets, Text Fields, Validation of EditText, Layouts, [dip, dp, sip, sp] versus px
Preferences: Shared Preferences, Preferences from xml
Menu: Option menu, Context menu, menu from xml, menu via code
Intents: Explicit Intents, Implicit intents
UI design: Time and Date, Images and media, Android Adapter and ListView, Composite, Alert Dialogs and Toast, Popup, Fragments, Navigation drawer

Tabs, Tab Activity Styles & Themes: styles.xml, drawable resources for shapes, gradients (selectors), style attribute in layout file, Applying themes via code and manifest file	
Content Providers: SQLite Programming, SQLite Open Helper, SQLite Database, Cursor, Reading and updating Contacts, Reading bookmarks	

Reference Books

1. Joseph Annuzzi Jr, Lauren Darcey, Shane Condor, "Advanced Android Application Development, Developers Library", Pearson Education, 4th Edition (2015)
2. Lauren Darcey, Shane Condor, "Android, Wireless Application Development",Pearson Education, 3rd Edition.
3. Paul Deitel, Harvey Deitel, Alexander Wald, "Android 6 for programmers, An AppDriven Approach", Pearson Education
4. Rap Payne, "Beginning App Development with Flutter: Create Cross-Platform Mobile Apps", Apress (2019)

Course Contents and Lecture Schedule

Sl No	Topic	No. of Lectures (Hours)
1	Fundamentals – Basic building blocks	3
2	Application structure, layout and resources	3
3	Android Virtual Device, Activity Lifecycle	3
4	Basic UI Design and EditText Validation	4
5	Shared Preferences, RelativeLayout, FrameLayout, GridLayout and Preferences from xml	9
6	ArrayAdapter, ListView and Exception handling	3
7	Various Menu options	3
8	Explicit and Implicit Intents	3
9	Images and media, Dialogs, Spinner component, Popups, Fragments, Navigation drawer	6
10	Applying themes and styles .xml	3
11	SQLite Programming	6

ST.JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

MCA Integrated

SEMESTER 10

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
24SJINMCA502	COMPREHENSIVE VIVA	VIVA	-	-	-	6

Preamble: Comprehensive Viva intends to assess the knowledge gained by a student in the core courses of this programme and to make the student aware of his/her knowledge level and where he/she stands after completing this programme. This course will help the student in preparing for comprehensive examinations and improve the confidence in answering questions in objective mode.

Prerequisite: Thorough knowledge in all the courses he/she learned during this programme.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	K Level
CO 1	Articulate the concepts in the core courses learned through this programme.	K2
CO 2	Attend technical interviews with confidence.	K2
CO 3	Interpret questions and answer them with clarity.	K2
CO 4	Make use of the concepts learned through this programme in future.	K3

Mapping of course outcomes with program outcomes

Mapping of course outcomes with program outcomes								
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2		2	2	2	2	3
CO2	3	2	1	2	3		3	3
CO3	3	2			2		2	2
CO4	3	2	2	2		2	3	3

Mark distribution

Total Marks	CIE	ESE
100	-	100

Guidelines:

- Viva shall be conducted by a panel of examiners consisting of:
  - 1. Head of the department
  - 2. A senior faculty in the department.
  - 3. External examiner appointed by the competent authority.

- Viva shall be conducted for each student for a minimum of 20 minutes.

- Knowledge level of the student shall be assessed on the following topics.

  - 24SJINMCA201 Computer Organization
  - 24SJINMCA206 Operating Systems
  - 24SJINMCA308 Design & Analysis of Algorithms
  - 24SJINMCA407 Advanced Software Engineering
  - 24SJINMCA402 Advanced Database Management Systems
  - 24SJINMCA404 Advanced Computer Networks
  - 24SJINMCA501 Data Science & Machine Learning
  - Code snippets in Java or Python to solve simple problems.
  - Technologies used in the project work.
  - Recent developments in the field of computer science.

24SJINMCA504	SEMINAR	CATEGORY	L	T	P	CREDIT
		SEMINAR	-	-	2	2

Preamble:	This course intends to enable the students to gain knowledge in any of the technically relevant current topics on Computer Science or Information Technology, and to acquire confidence in presenting the topic and preparing a report
Prerequisite:	Nil

Course Outcomes: After the completion of the course the student will be able to:		**K Level**
CO1	Annotate the ideas presented in technical papers	K2
CO2	Comprehend a concept by referring different technical documents	K2
CO3	Prepare technical documents	K2
CO4	Present a topic before an audience	K2
CO5	Interact with the audience	K2

Mapping of course outcomes with program outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	1	1	1	1		2
CO2	3	3	2	2	1	1		2
CO3	3	3	2	2	2	2	1	3
CO4	2	2	2	3	3	2	1	3
CO5	1	2	1	2	3	2	1	3

Mark Distribution

Total Mark	: 50 Marks
Continuous Evaluation Marks	: 20 Marks
End Semester Examination	: Nil

Assessment Criteria

Scope and relevance of topic	: 20%
Quality of Presentation slides	: 10%
Presentation Skills	: 30%
Knowledge in the topic	: 20%
Report	: 20%

Marks Division

Evaluation by Faculty Guide	: 20 Marks
Evaluation by Faculty Committee	: 30 Marks
Total	: 50 Marks

Guidelines

- Students shall conduct detailed study on a technically relevant current topic in Computer Science / Information Technology under the supervision of a Faculty Guide and present it as a seminar at the end of the study.
- The study may be conducted on
    - Articles published in reputed journals/conference proceedings
    - Recent development in Computer Science / Information Technology
    - Recent research and development activity in a research lab
    - Latest software tool or framework
- Students shall submit an abstract on an identified topic and get prior approval from the Faculty Guide before the study begins.
- The student shall submit a seminar report, based on the study and their findings. The report shall not be a reproduction of the original paper or manual.
- The study and its findings shall be presented in the class taking a duration of 15-20 minutes.

- LaTeX or an equivalent tool shall be used for preparing Presentations and Seminar Report.
- Students shall be encouraged to publish their study in journals and due credit shall be given to such students.
- A committee of three senior faculty members shall be constituted by the head of the department and the seminar presentation shall be evaluated by that committee.

24SJINMCA506	MAIN PROJECT	CATEGORY	L	T	P	CREDIT
		PROJECT	-	-	27	12

Preamble: This project work aims to enable the students to apply the software engineering principles on a real software project, to make the students familiar with the stages of a deployment pipeline and to develop a software product using the latest software development methodology.

Prerequisite: Knowledge in software engineering principles and programming skills.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	K Level
CO 1	Identify a real-life project which is useful to society / industry	K2
CO 2	Interact with people to identify the project requirements	K3
CO 3	Apply suitable development methodology for the development of the product / project	K3
CO 4	Analyze and design a software product / project	K4
CO 5	Test the modules at various stages of project development	K5
CO 6	Build and integrate different software modules	K5
CO 7	Document and deploy the product / project	K3

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	1	2	2	2	2
CO2	3	2	2	3	2	2	2	2
CO3	3	2	2	3	2	1	1	2
CO4	3	2	2	3	2	1	2	3
CO5	3	2	1	3	1	2	1	3
CO6	3	2	2	3	2	2	1	2
CO7	3	2	2	3	1	1	1	3

Mark distribution

Total Marks	CIE	ESE
100	70	30

Marks Division

Continuous evaluation by Supervisor, Guide(s) and Scrum Master	30 Marks (Internal)
Evaluation by the Project Assessment Board	40 Marks (Internal)
Evaluation by the External expert	30 Marks (External)
Total	**100 Marks**

Guidelines:

- Students shall identify Real-Life Projects which are relevant and useful to the society or industry.
- The project shall be an individual project and must be done in-house. The student has to spend time in the lab for the project work. Attendance as per MCA regulations is applicable for submitting the project for final evaluation.
- However, in exceptional cases students shall be given permission to work on the project outside the campus and at the industry premises if the organization offering the project belongs to anyone of the following categories.
    - CMM Level 5 Certified Company
    - Publicly listed company in India
    - National Research Institute
    - Central / State Government Department
    - Project funded by the Central / State Government Agency
    - Companies approved by department internship scrutiny committee
- In such cases, the student is required to produce a letter from the organization before starting the project and an internship scrutiny committee constituted by the head of the department shall make the decision on permission. Industries and training institutes that offer project work for a fee shall not be permitted.
- Students shall submit project synopsis and get prior approval from the Project (Faculty) Supervisor before the project work begins.
- If there is a customer for the project then he/she will be the Product Owner (External Guide) and a faculty from the department will be the Internal Guide. If there is no such customer then the Internal Guide himself/herself shall act as the Product Owner.
- A faculty / technical staff shall act as the Scrum Master to continuously monitor the project development. Periodic meetings, of less than 15 minutes, at the convenience of

the Scrum Master are to be highly encouraged. Ensure such meetings occur once in three days.

- The student shall maintain a Scrum Book (Rough Record) which has to be divided into 4 parts – (i) Product Backlog (ii) Database & UI Design (iii) Testing & Validation and (iv) Details of Versions. Make dated entries in the corresponding part at regular intervals. The corrections and comments from Product Owner and Scrum Master should be clearly indicated with the Date.
- Test Driven Development methodology may be practiced for the project development. BugZilla, BackLog or any such tool may be used for Bug Tracking.
- Git shall be used for Version Control and Git commit history may be verified as part of project evaluation.
- LaTeX or an equivalent tool shall be used for preparing Presentations and Project Report.
- Students shall be encouraged to publish their work in journals and due credit shall be given to such students.
- For the externally done projects, periodic confidential progress report and attendance statement shall be collected from the External Guide and be reviewed by the Project Supervisor.
- Set a sprint as two weeks, ensure biweekly reviews. A review shall not exceed 30 minutes. A demo to the Product Owner (Project Guide) is mandatory in every review.
- Interim evaluations of the project's progress shall be conducted by a Project Assessment Board as part of internal assessment. Two such evaluations are desirable. Scrum reviews shall not be sacrificed for such presentations.
- The Project Assessment Board shall be constituted by the Head of the Department with the following five members.

Chairman: 1. Head of the Department

Members: 2.Project supervisor/s of the student

3.One faculty member from the Department

4. One faculty member from a sister Department

5. An external expert, either from an academic/research institute or Industry. (For the externally done projects, the external guide

- At the end of the semester, two evaluations shall be there on the entire project development activities. First an internal evaluation by the Project Assessment Board and second an external evaluation by an External Examiner.
- An External Examiner either from an academic institute or industry shall be appointed by the Authority concerned for the External Evaluation.

Week	Schedule
(May be scheduled in line with the College academic calendar)	
1	Selection of Topic, Submission of project synopsis and getting approval Meeting of Development Team including Scrum Master with Product Owner (Project Guide)
2	Commencement of the Project.
4	First Sprint release and Scrum Review by the Product Owner (Project Guide)
6	Second Sprint release and Scrum Review by the Project Guide First interim evaluation by the Project Assessment Board
8	Third Sprint release and Scrum Review by the Project Guide
10	Fourth Sprint release and Scrum Review by the Project Guide
11	Second interim evaluation by the Project Assessment Board
12	Fifth Sprint release and Scrum Review by the Project Guide
13	Submission of project report, with Scrum Book Final project presentation Evaluation by the Project Assessment Board
14	Final evaluation by the External Examiner.

Department of

Computer Applications

Vision

To emerge as a center of excellence in the field of computer education with distinct identity and quality in all areas of its activities and develop a new generation of computer professionals with proper leadership, commitment, and moral values.

Mission

- Provide quality education in Computer Applications and bridge the gap between the academia and industry.
- Promote innovation, research, and leadership in areas relevant to the socio-economic progress of the country.
- Develop intellectual curiosity and a commitment to lifelong learning in students, with societal and environmental concerns.

ST.JOSEPH'S COLLEGE OF ENGINEERING AND TECHNOLOGY, - PALAI -

AUTONOMOUS

Vision

Developing into a world class, pace setting institute of Engineering and Technology with distinct identity and character, meeting the goals and aspirations of the society.

Mission

- To maintain a conducive infrastructure and learning environment for world class education.
- To nurture a team of dedicated, competent and research-oriented faculty.
- To develop students with moral and ethical values, for their successful careers, by offering variety of programs and services.